
UML Documentation

for

Automated Lab Program Evaluator

Version 1.0 approved

**Prepared by
Ashwin Joisa - 16CO104
Praveen Raj - 16CO115**

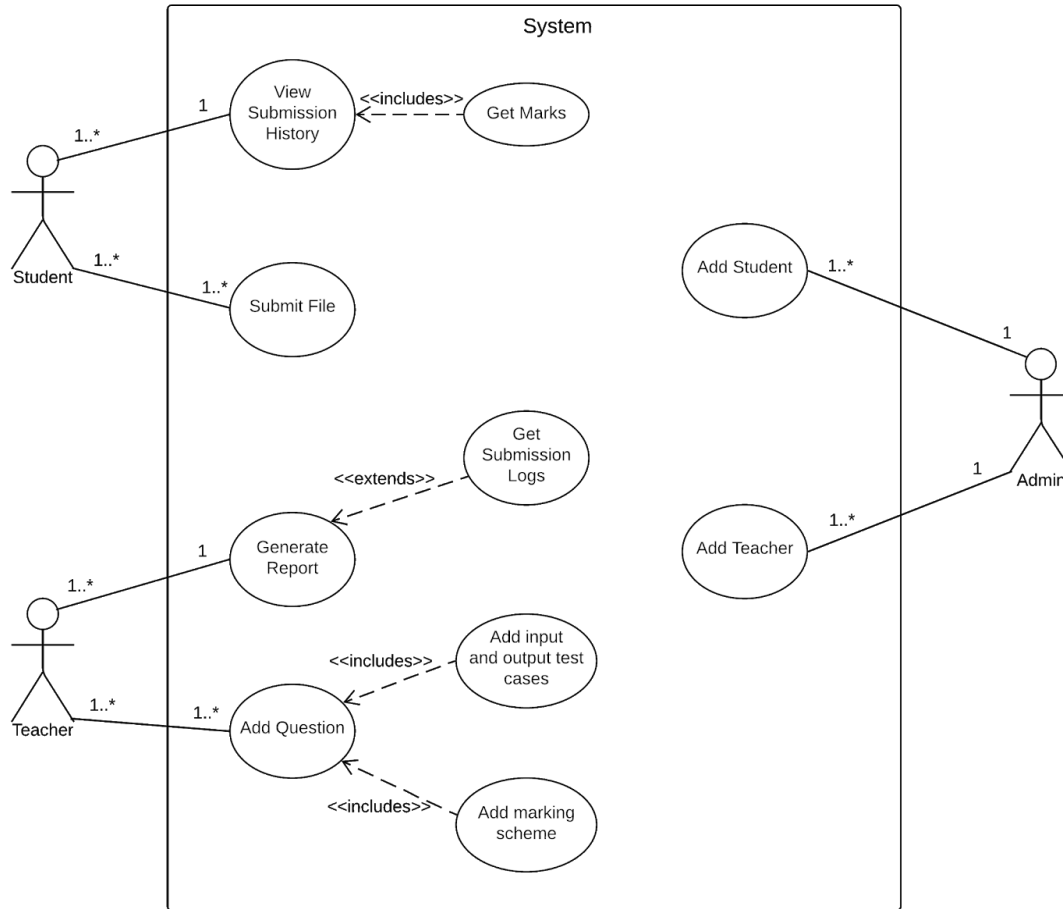
NITK, Surathkal

25/02/2018

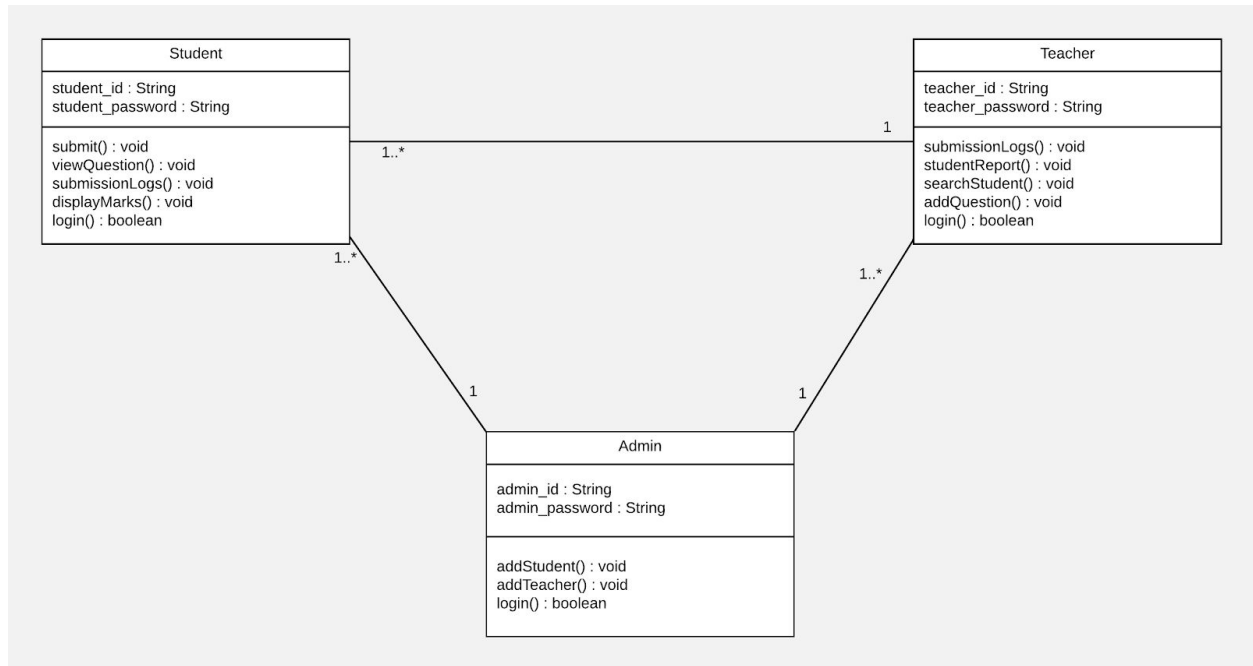
Contents

1. Use Case Diagram	3
2. Class Diagram	4
3. Activity Diagram	5
4. Sequence and Collaboration Diagram	6
5. State Machine Diagram	10
6. Component Diagram	12
7. Deployment Diagram	12

1. Use Case Diagram



2. Class Diagram

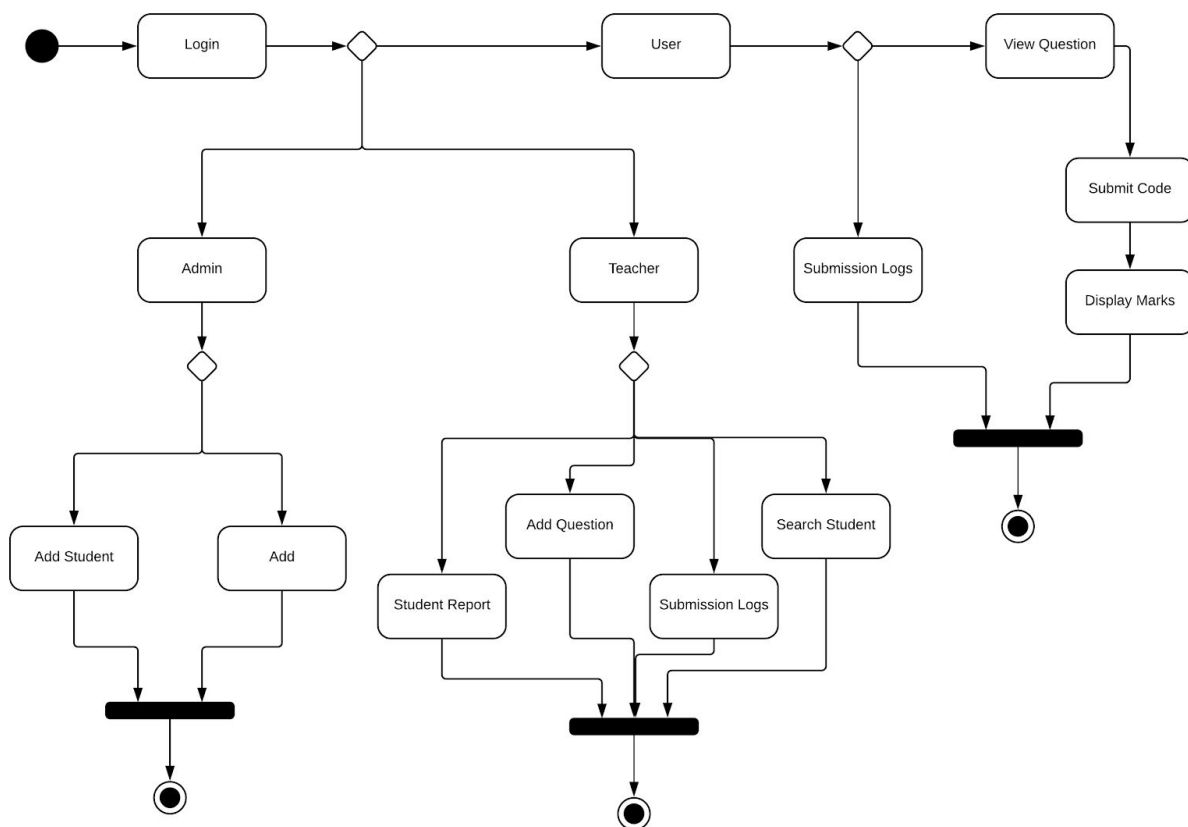


3. Activity Diagram

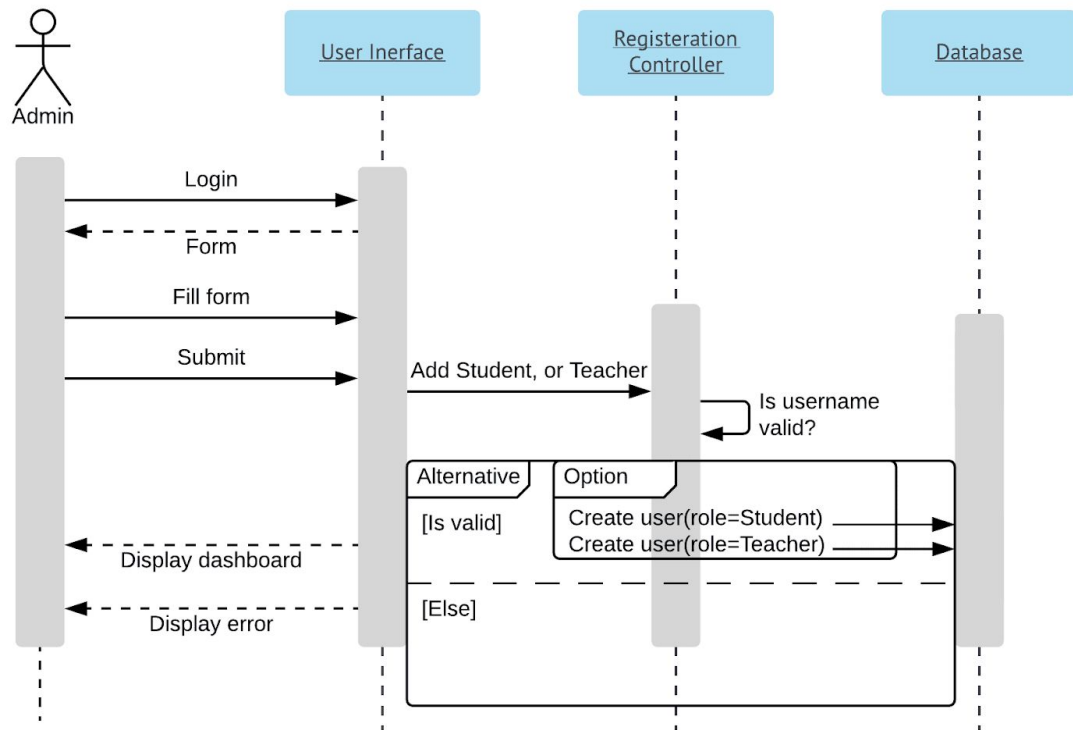
Activity is a parameterized behavior represented as coordinated flow of actions.

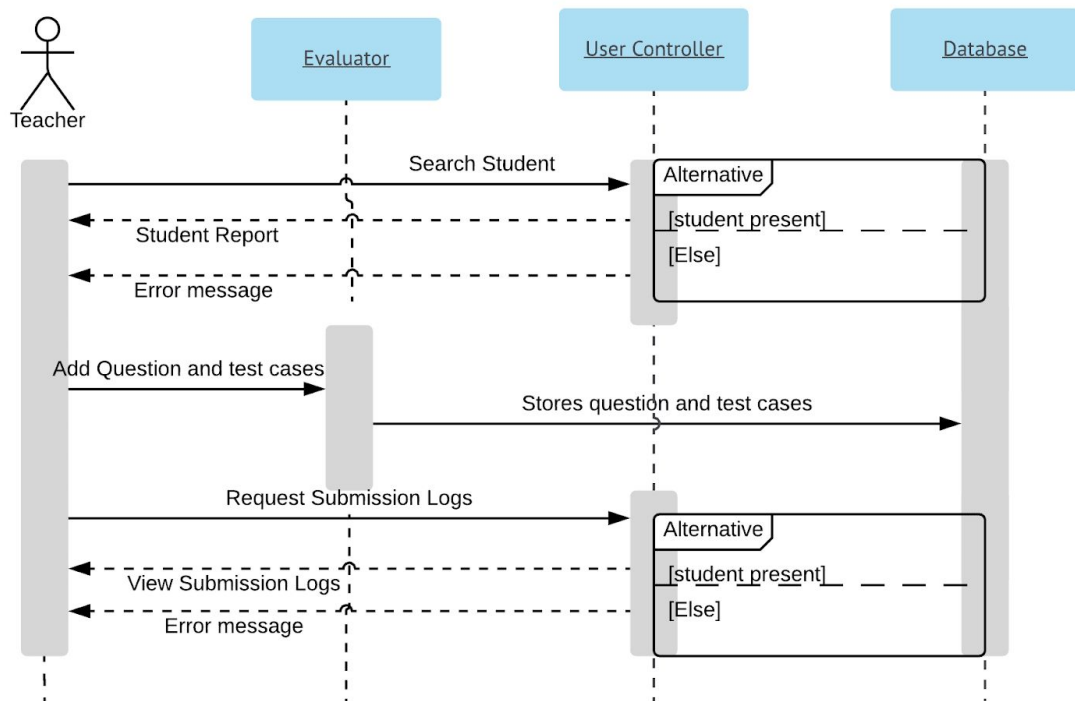
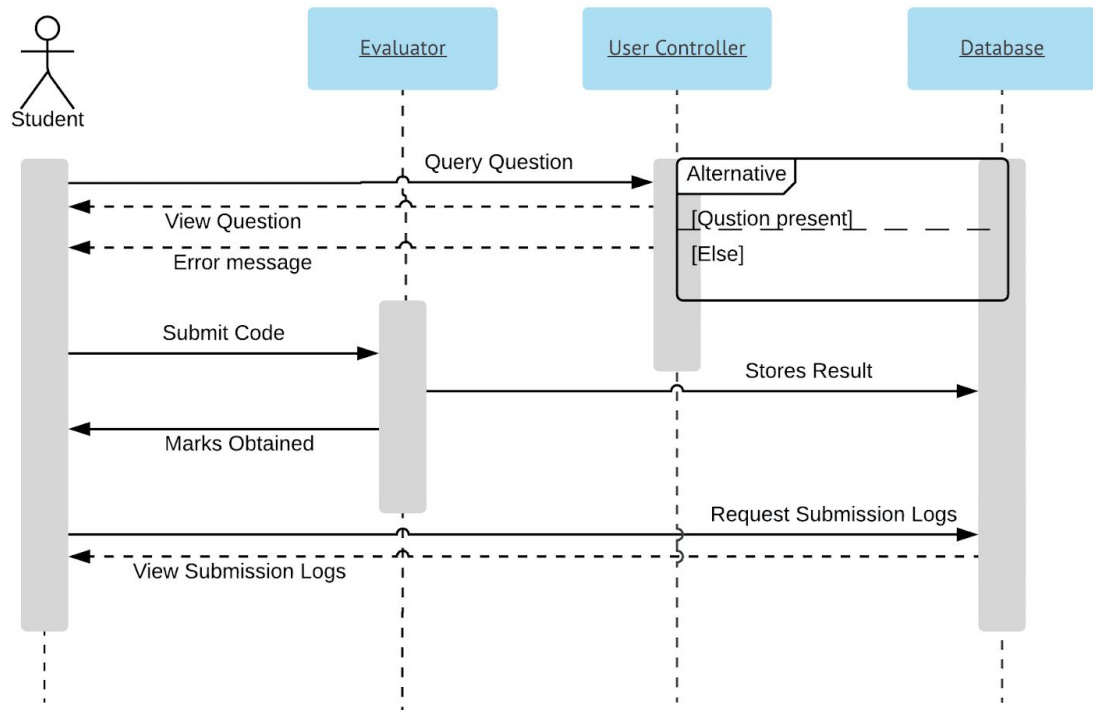
The **flow of execution** is modeled as activity nodes connected by activity edges. A node can be the execution of a subordinate behavior, such as an arithmetic computation, a call to an operation, or manipulation of object contents. Activity nodes also include flow of control constructs, such as synchronization, decision, and concurrency control. Activities may form invocation hierarchies invoking other activities, ultimately resolving to individual actions. In an object-oriented model, activities are usually invoked indirectly as methods bound to operations that are directly invoked.

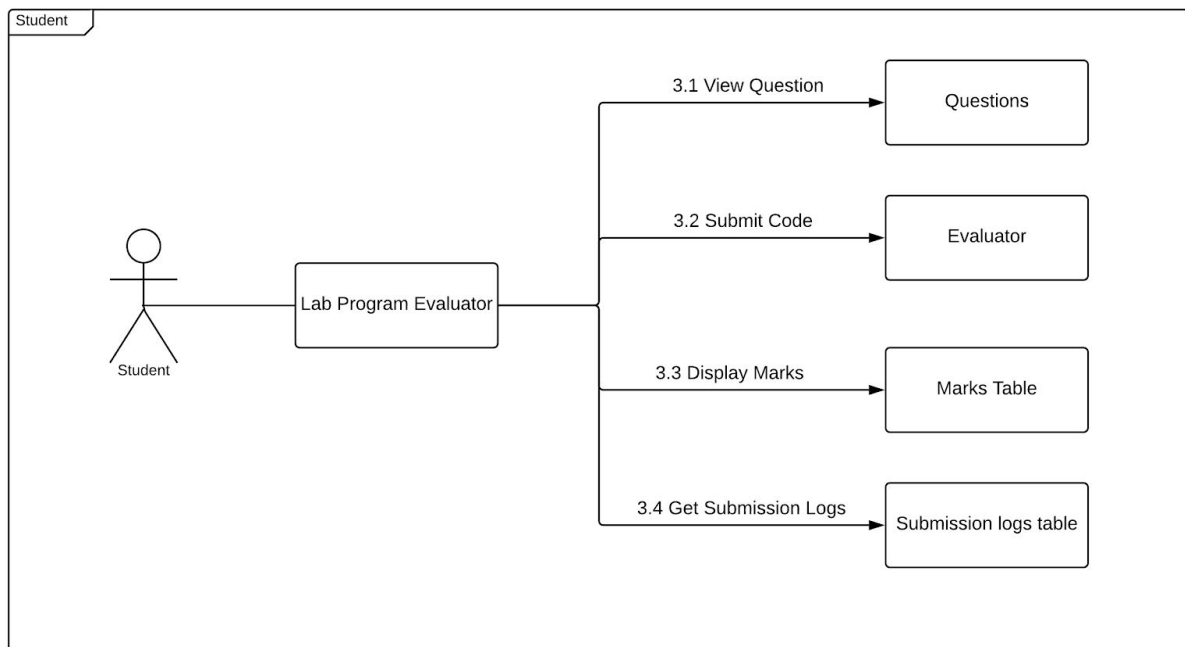
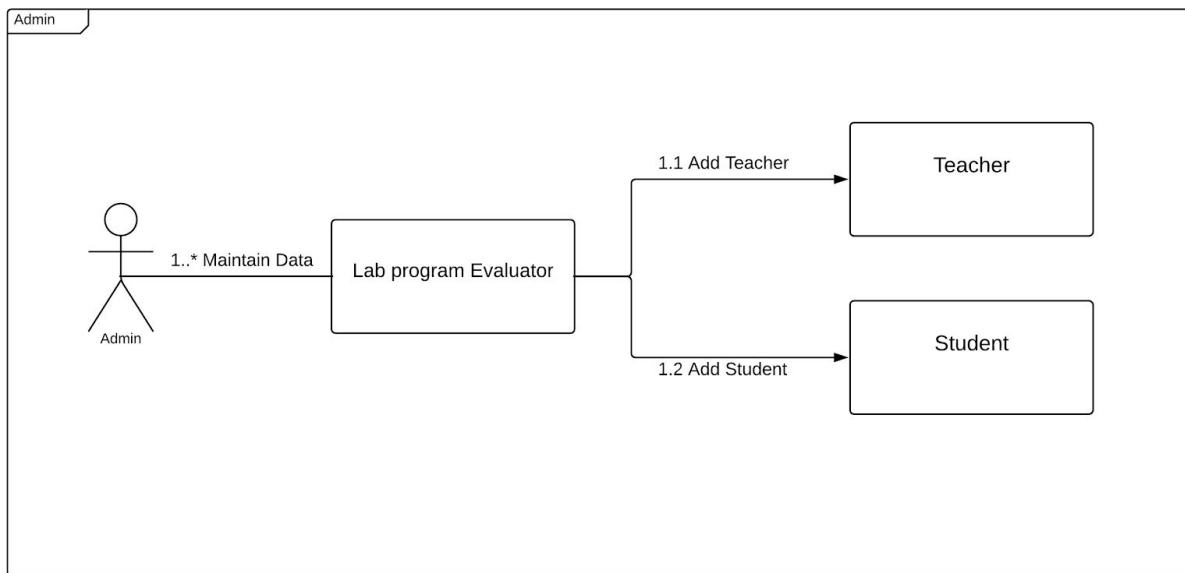
Once the system starts, the admin/teacher/student can login to the system, and the credentials of the user are authenticated. Student can view a question, submit code for the same, and receive the marks obtained. He/she can also the submission history. Teacher can add questions to the database, view submission logs, student's marks, and search for a particular student. Admin has privileges to add a student or teacher to the database.

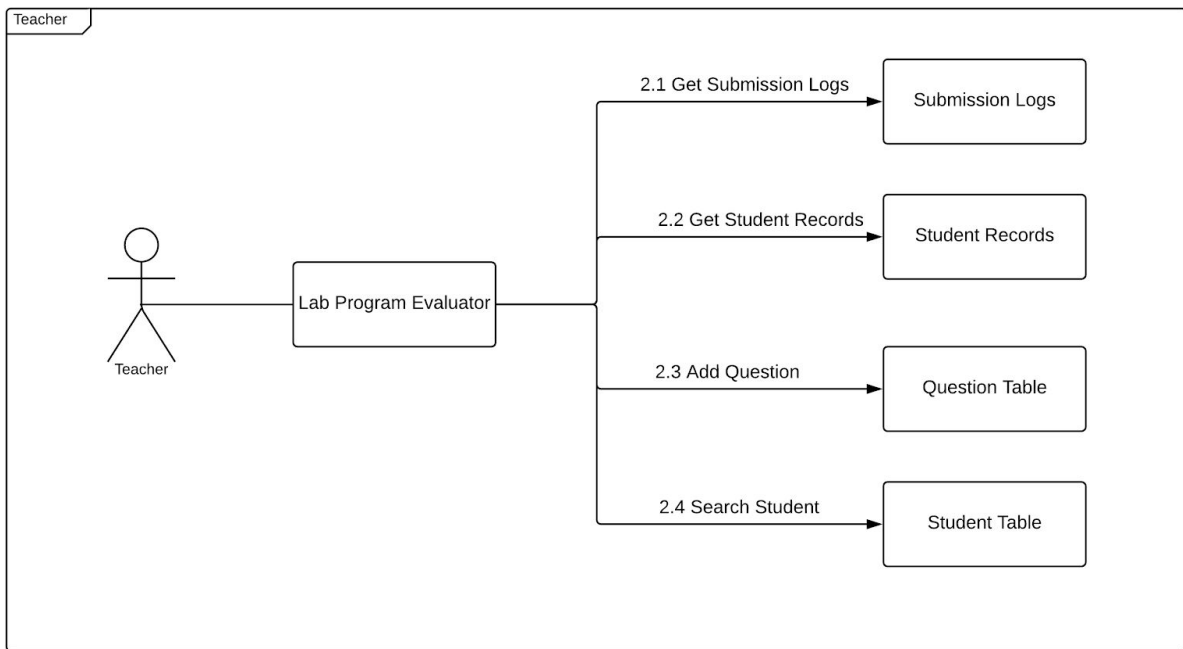


4. Sequence and Collaboration Diagram

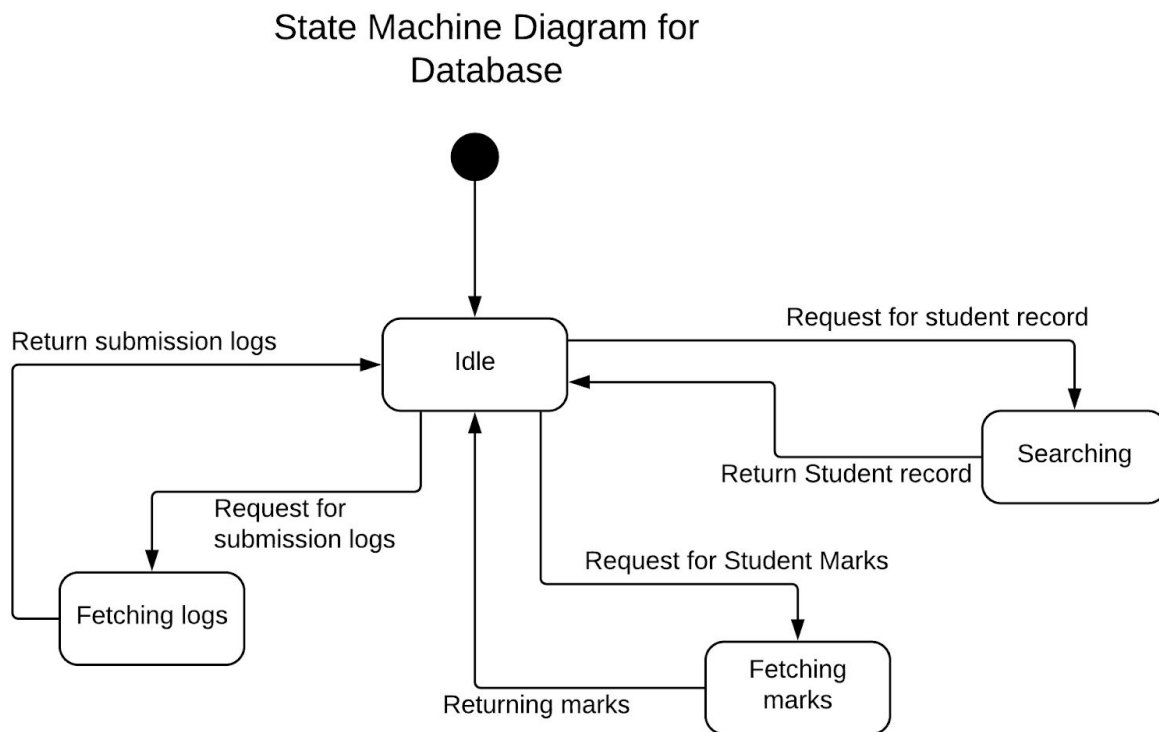




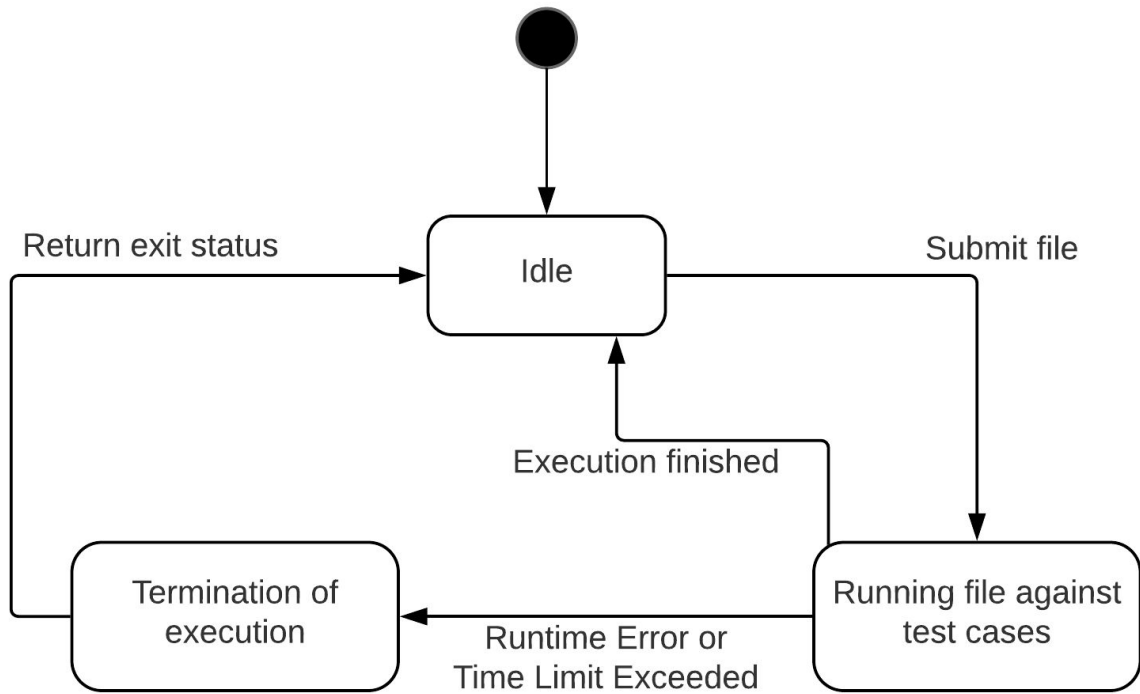




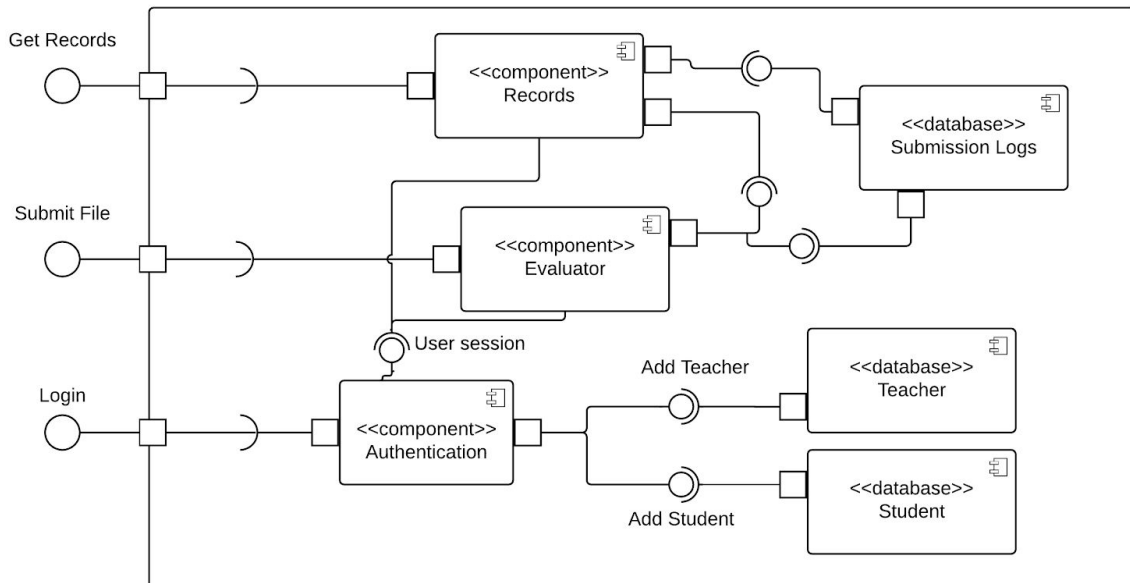
5. State Machine Diagram



State Machine for Evaluator



6. Component Diagram



7. Deployment Diagram

