Event Management System : Design and implementation with an application

Maj Amit Pathania(163054001)
Dharmendra (163050032)

April 2017

Contents

1 Problem Description **3**

2 Viewpoints **4**
2.1 Functional viewpoint 4
2.2 Information viewpoint 5
2.3 Concurrency viewpoint 8
2.4 Development viewpoint 10
2.5 Deployment viewpoint 12
2.6 Operational viewpoint 13

3 Perspectives **14**
3.1 Security perspective 14
3.2 Performance perspective 16
3.3 Availability and resilience perspective 17

List of Figures

1 UML component diagram for functional viewpoint 4
2 ER diagram between user and event 5
3 ER diagram as Subscriber and publisher 5
4 UML class diagram for information viewpoint 6
5 Data flow diagram diagram for information viewpoint 7
6 Thread based concurrency model using UML 8
7 Statechart for query processing 9
8 Development model for server using UML module structure . 10
9 Development model for client using UML module structure . 11
10 Deployment viewpoint using UML 12
11 Attacktree for security perspective 15
12 UML model for performance perspective 16
13 Recovery diagram 18

List of Tables

1 Sensitive Resource Identification 14
2 Access Control list 14
3 Incident recovery Table for availability perspective 17

1 Problem Description

We aim to create a event management system using C++ where client conencts to server suing TCP connection. Server has mysql server which is used to authenticate the users and once authenticated, user can create events and events will be pushed to user as per his interests/topics.

2 Viewpoints

2.1 Functional viewpoint

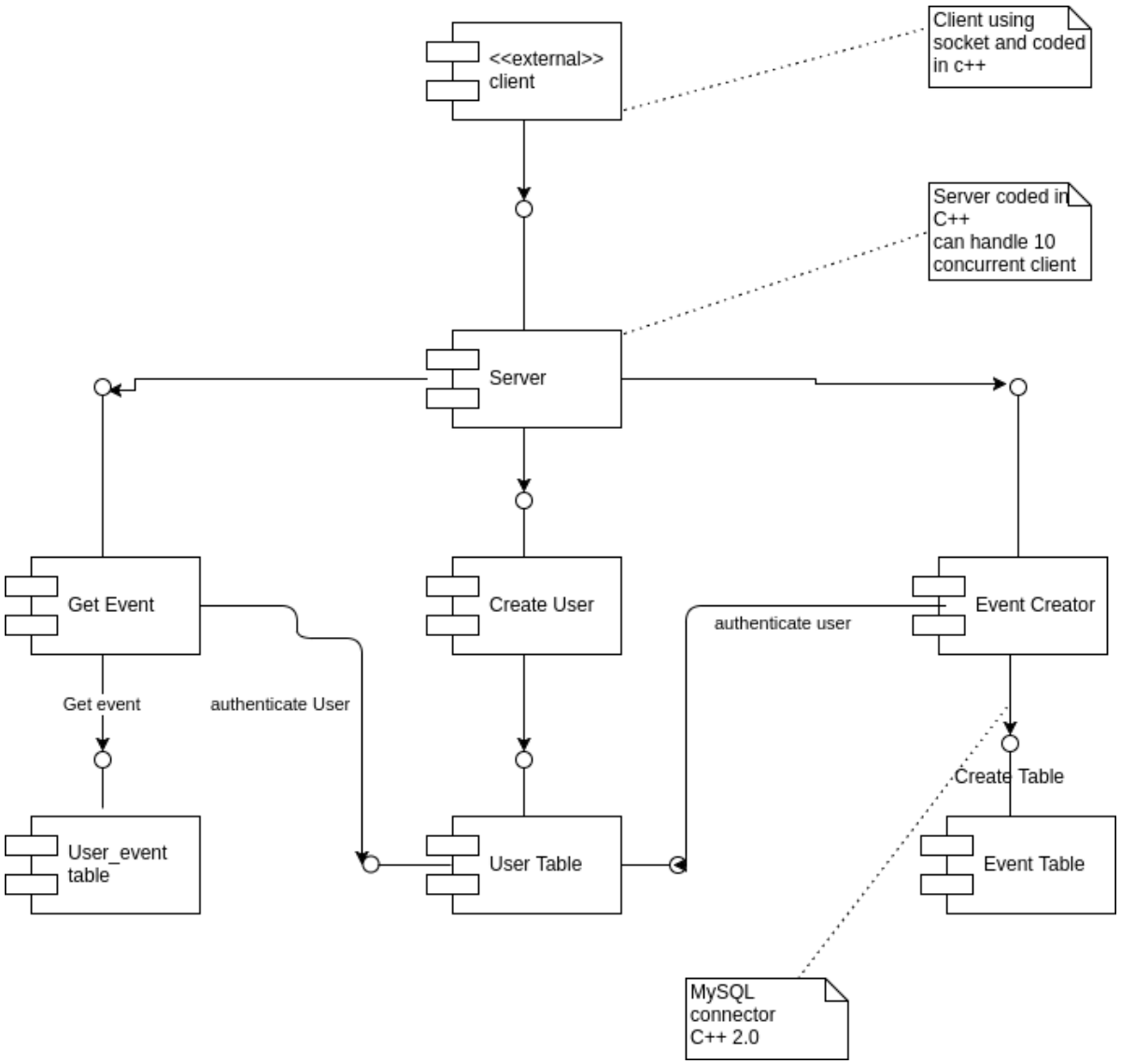

Figure 1: UML component diagram for functional viewpoint

2.2 Information viewpoint

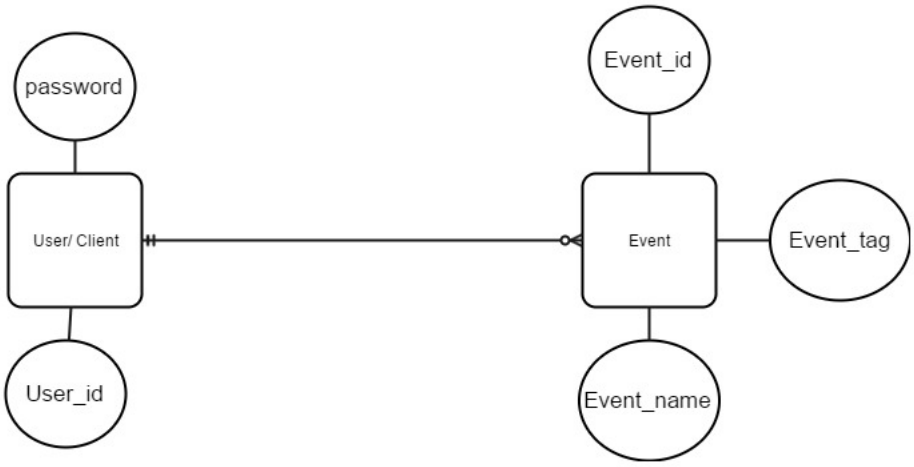

Figure 2: ER diagram between user and event

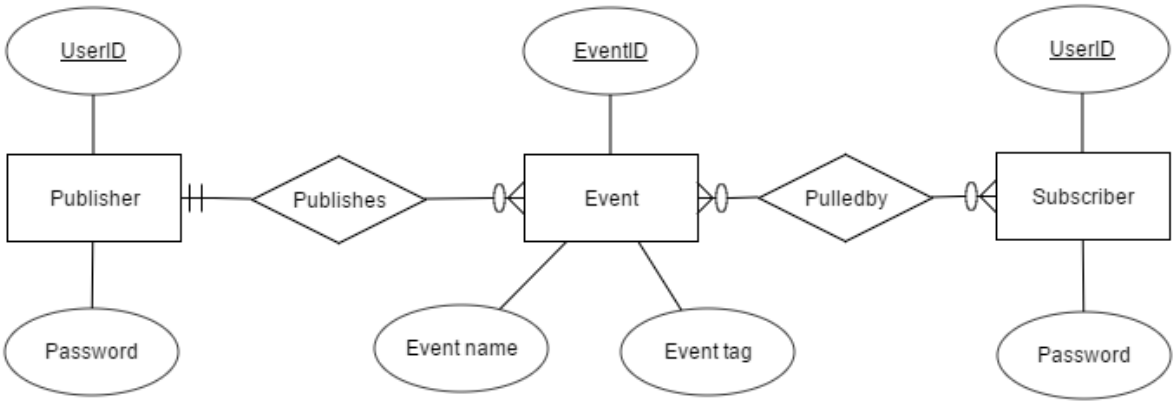

Figure 3: ER diagram as Subscriber and publisher

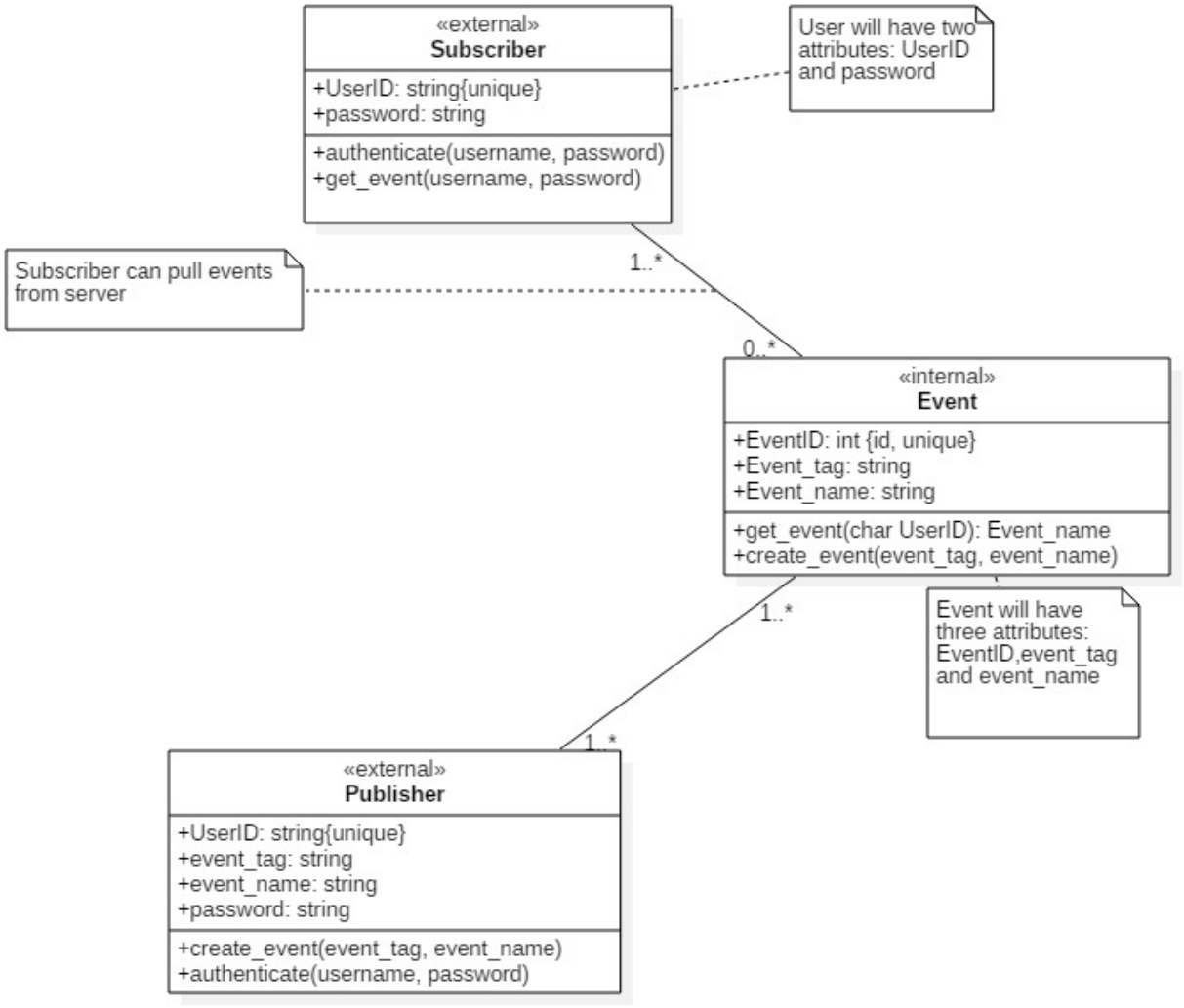

Figure 4: UML class diagram for information viewpoint

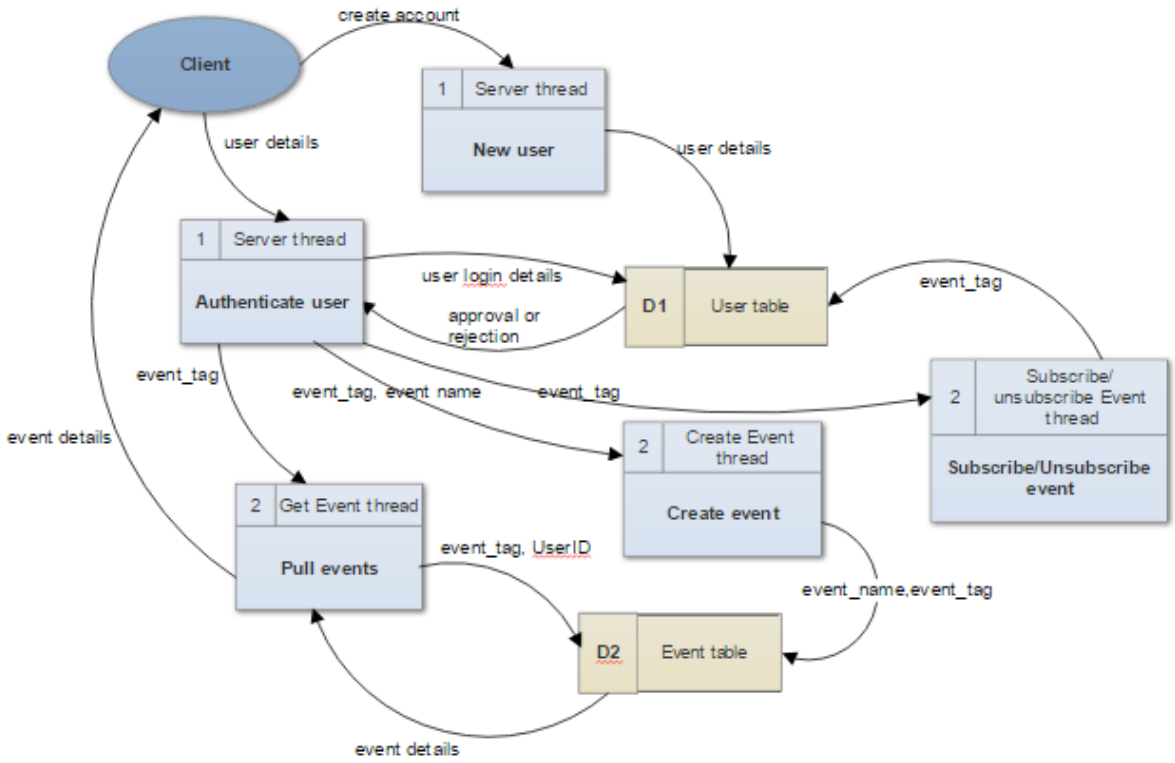

Figure 5: Data flow diagram diagram for information viewpoint

2.3 Concurrency viewpoint

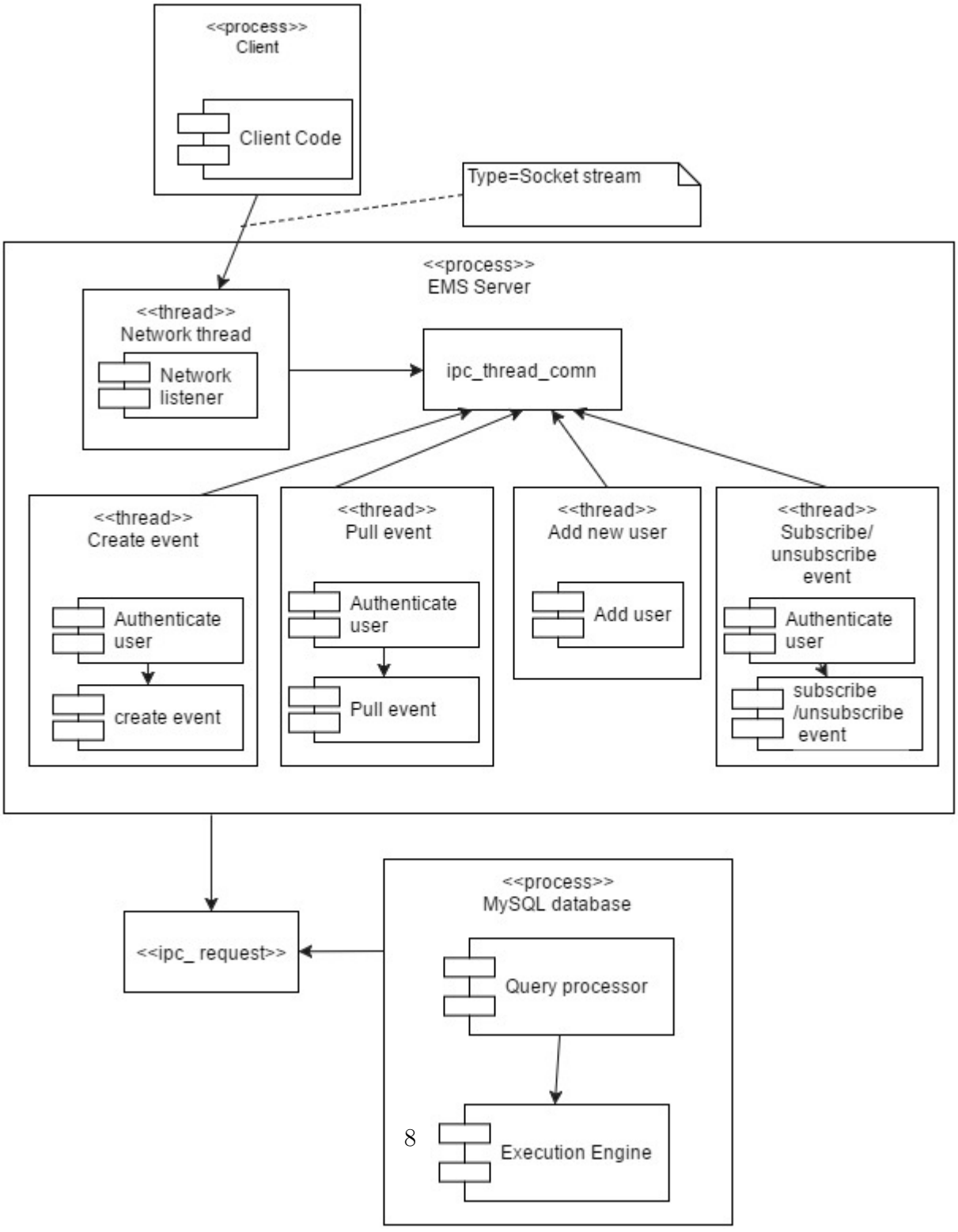

Figure 6: Thread based concurrency model using UML

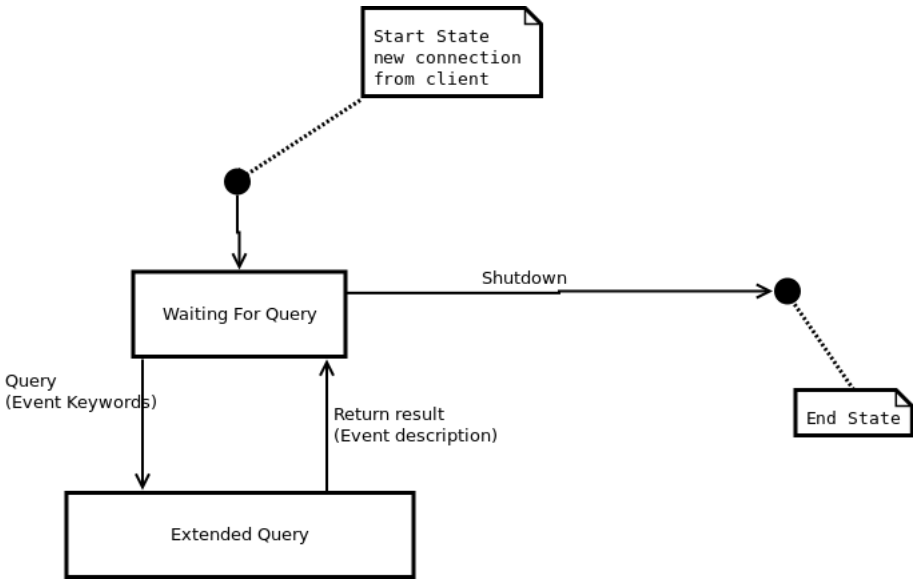

Figure 7: Statechart for query processing

2.4 Development viewpoint

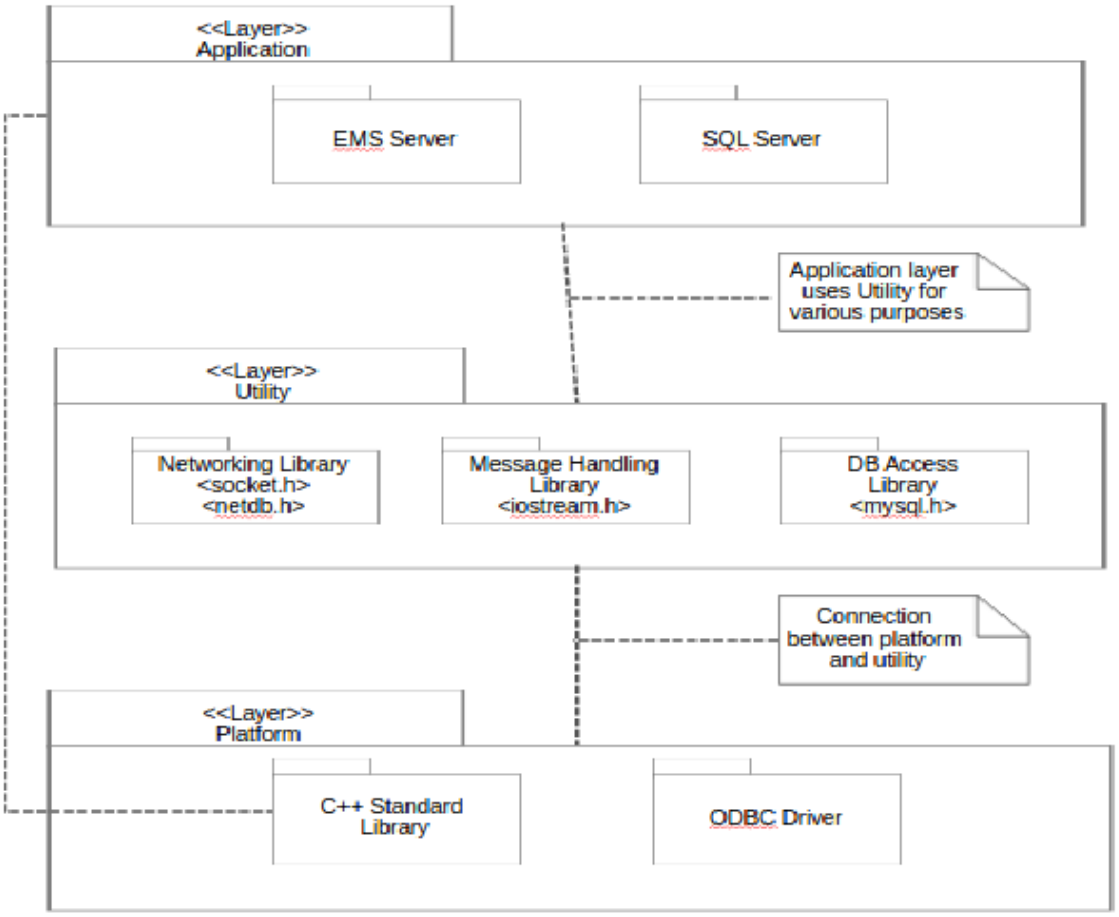

Figure 8: Development model for server using UML module structure

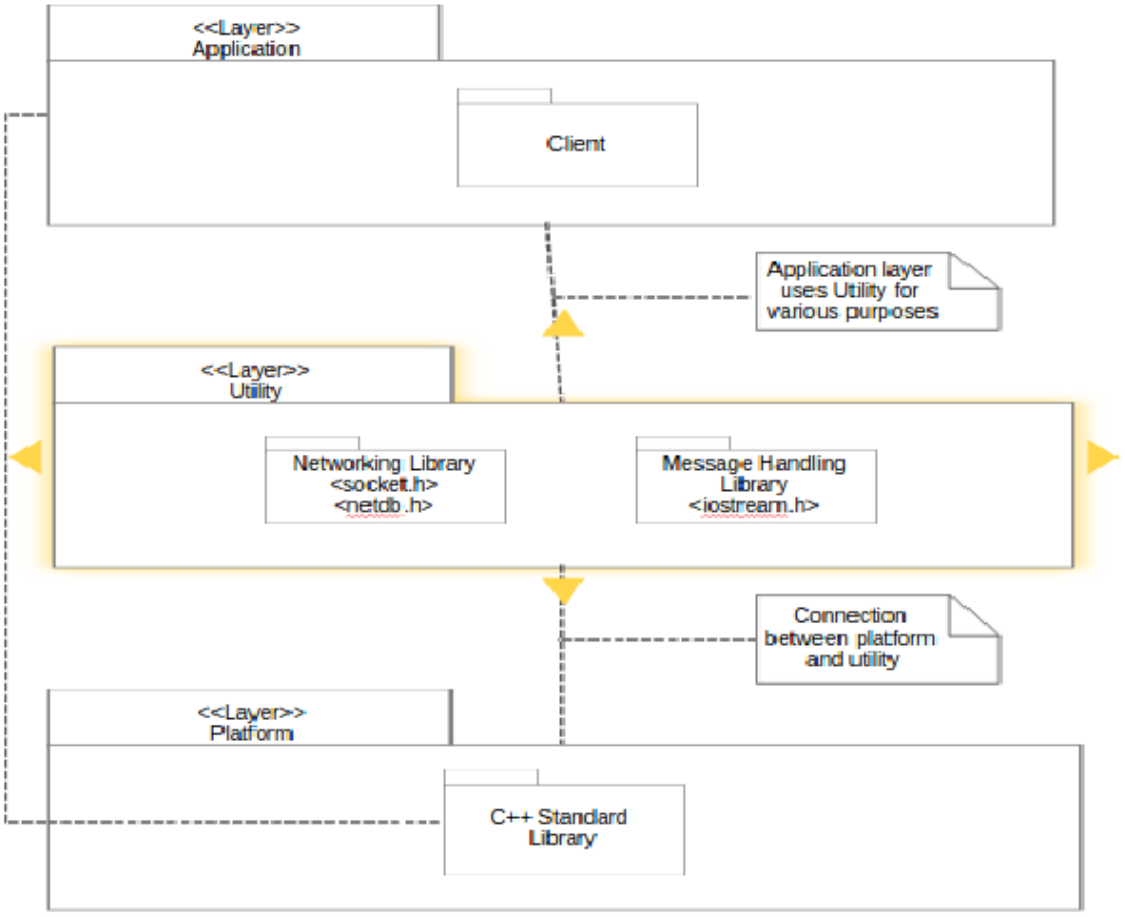

Figure 9: Development model for client using UML module structure

2.5 Deployment viewpoint

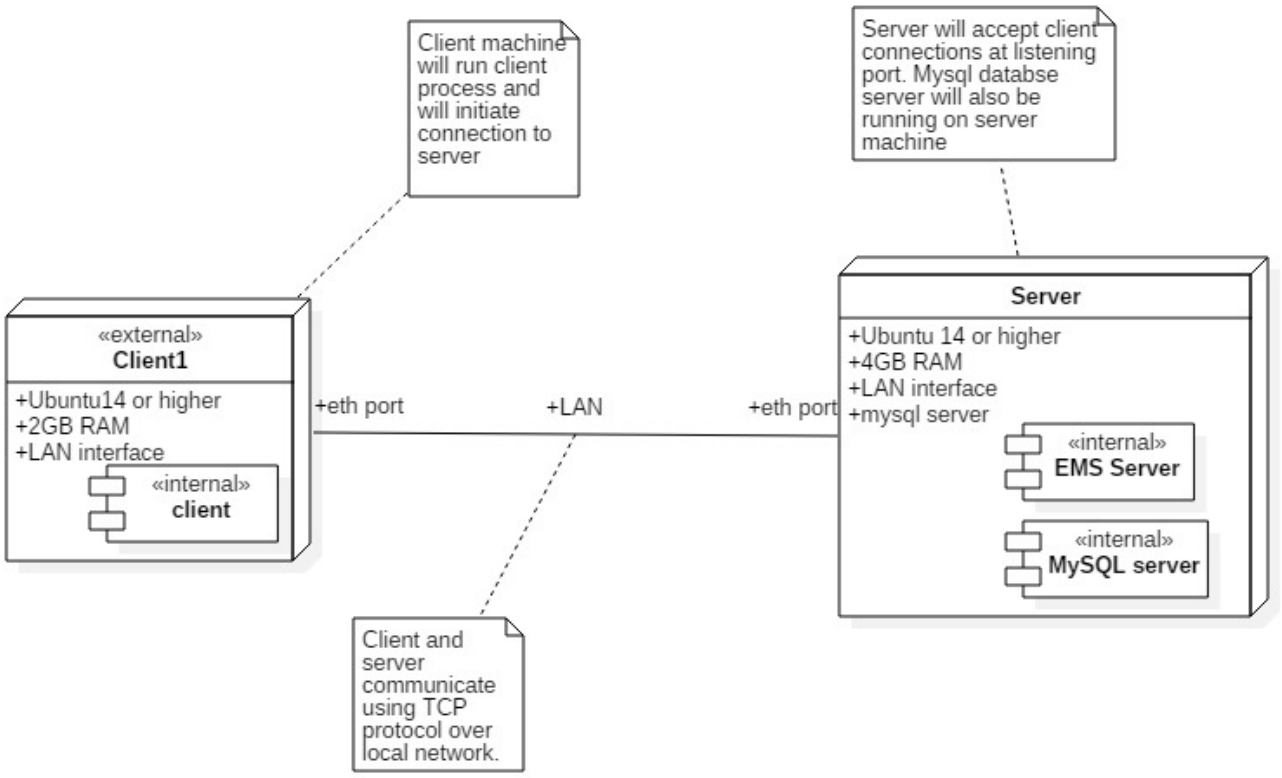

Figure 10: Deployment viewpoint using UML

2.6 Operational viewpoint

Installation model helps in moving a system from its development environment to its production environment.It explains how system can be installed.

- **Installation Groups**

Server.o

Ubuntu 14.0 or above Server

Installation using "make" script

Client.o

Ubuntu 14.0 or above desktop client

Installation using "make" script.

Mysql Database server

Install using command on server: sudo apt-get install mysql-server

mysql.h header file

Install using command on server: sudo apt-get install libmysqlclient-dev

- **Dependencies**

Server.o depends on mysql.h and mysql database server.

Client.o and server.o depends on gcc compiler and standard gcc libraries.

- **Constraints**

Server.o : A access to mysql database is required by server. Hence, adequate privileges should be granted to server

3 Perspectives

3.1 Security perspective

Resource / Table name	**Sensitivity**	**Owner**	**Owner Access Control**
user	Personal information of value for identity theft or invasion of privacy	Client Management Group (Admin)	No direct access allowed for other users
Event	Defines what events have been created and its description; if changed maliciously, could harm the authenticity of the system.	Event management group (Registered users)	Access to events records permitted after authentication of client(user).

Table 1: Sensitive Resource Identification

	User	**Event**
Database Administrator	All with audit	All with audit
Registered Customer	All on own record only	Read-only operation to pull event and publish events
Unknown user	None. Can create new account.	None

Table 2: Access Control list

Attack Tree for the goal of extracting user account details from server

Goal: Obtain user account details.

1. Extract details from the system database.
- **1.1.** Access the database directly.
  - **1.1.1.** Crack/guess database passwords.
  - **1.1.2.** Crack/guess operating system passwords that allow database security to be bypassed.
- **1.2.** Access the details via a server management staff.
  - **1.2.1.** Bribe a database administrator (DBA) at server side.

2. Extract details from the client.
- **2.1.** Set up a dummy/attack server and intimate clients/users the new IP address and listening port to trick them into getting connected to server at new IP which asks user for authenticating and thereby fooling client to enter its username and password.
- **2.2.** Monitor the TCP traffic originating from client terminal using TCP packet analyser and copy user account details.

Figure 11: Attacktree for security perspective

3.2 Performance perspective

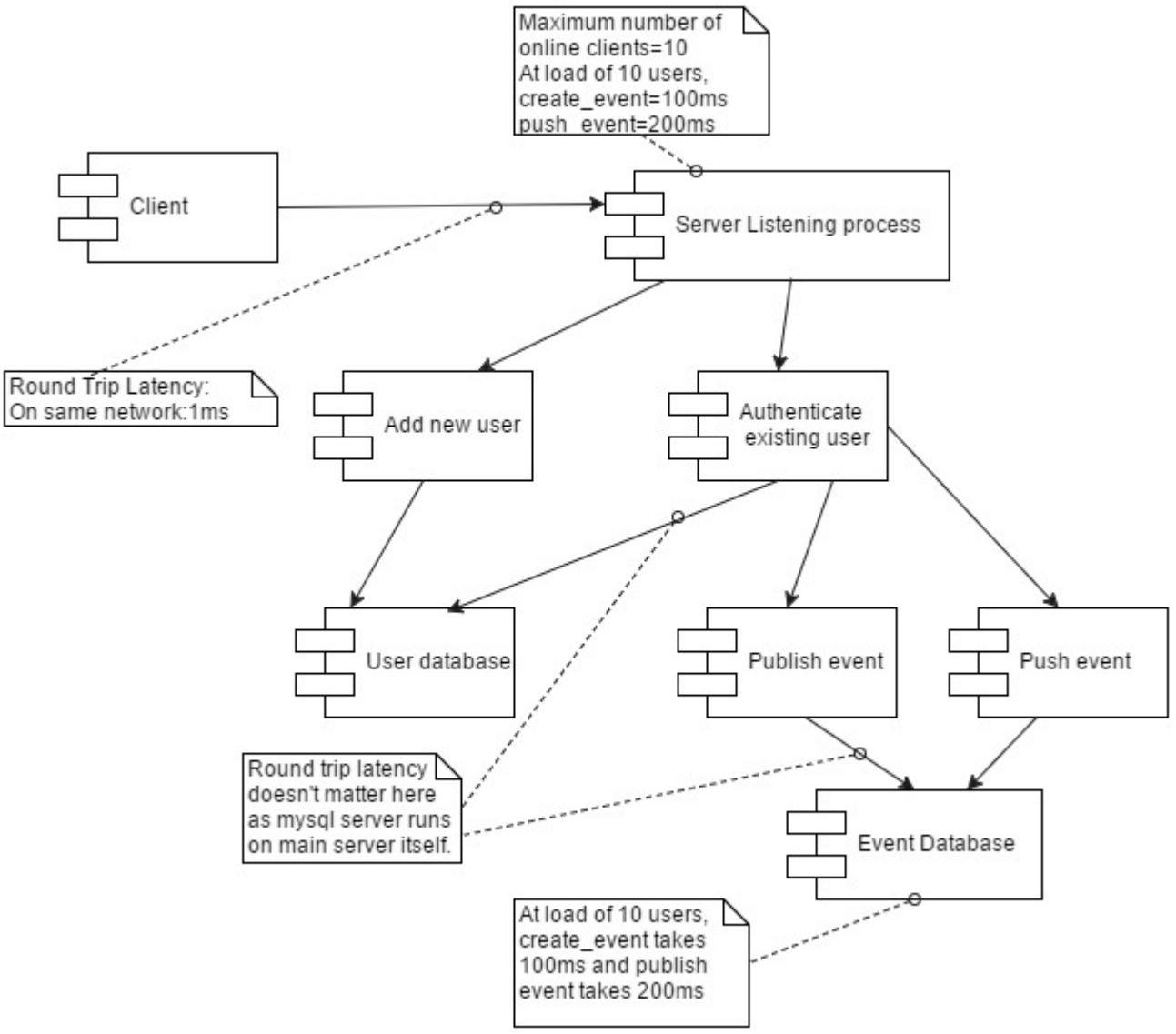

Figure 12: UML model for performance perspective

3.3 Availability and resilience perspective

Incident	Impact	Remedial Action	Time to Repair
Server Hardware (nondisk) failure	Reduced availability (throughput affected)	Replace the faulty component, and possibly reconfigure the hardware or the operating system	2 hours
Server Disk failure	Total unavailability (service offline)	Replace the faulty disk, and possibly recover data from backup	8-10 hours
Local network failure	Temporary service outage, in-flight transactions aborted.	Switch over to standby network Or repair the network after finding fault	5 mins to 1 hour
Operating system crash	Temporary service outage, normal availability within 5–10 minutes	Reboot; although crash may be some other problem (such as faulty memory or disk) that needs to be addressed	5 mins for reboot
Mysql server crash	Temporary service outage	Restart the service	2-5 mins to restart the service
Segmentation fault	Temporary service outage	Restart the application	1 min

Table 3: Incident recovery Table for availability perspective

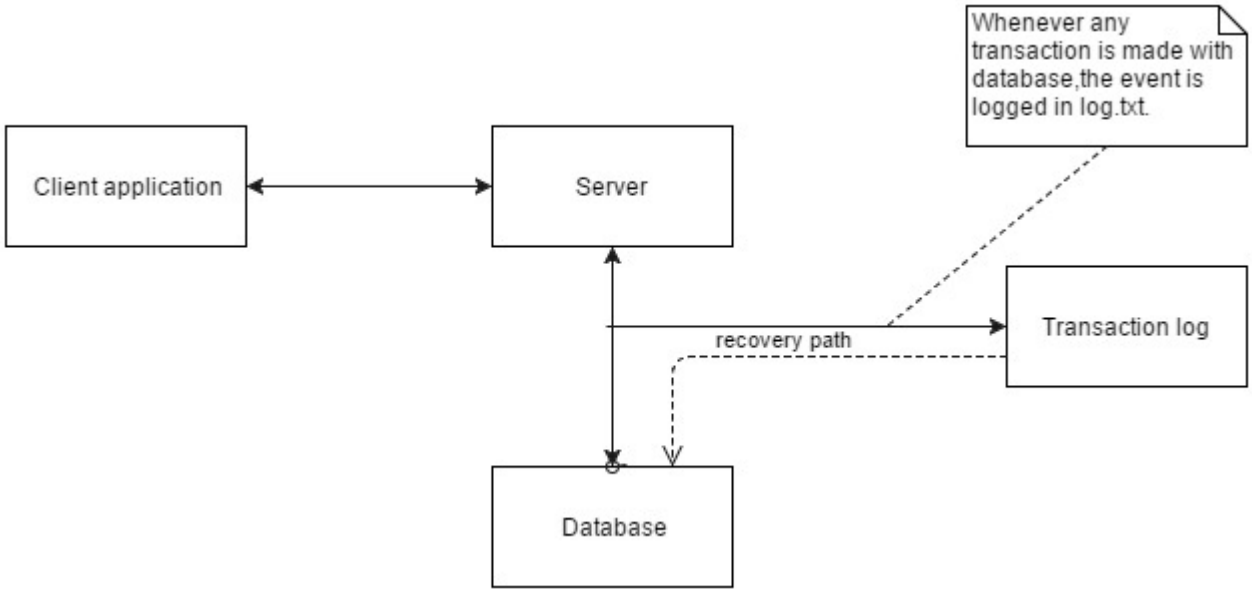

Figure 13: Recovery diagram