

U.B.A. FACULTAD DE INGENIERÍA

Departamento de Computación

Técnicas de programación concurrente I 75-59 Informática

TRABAJO PRÁCTICO N°2

ConcusQL

Curso 2015 - 1er Cuatrimestre

Grupo N°7		
APELLIDO , Nombres		N°Padrón
MARTINEZ	Gaston Alberto	91383
MERLO SCHURMANN	Bruno Javier	91818

Observaciones:

Índice

1. Análisis del problema	2
2. División del proyecto en procesos	2
3. Comunicación entre procesos	2
4. Mecanismos de concurrencia utilizados	2
4.1. Comunicación cliente-servidor	2
4.2. Logger	2
5. Diagrama de clases	3
6. Diagrama de casos de uso	5

1. Análisis del problema

Se pide realizar un gestor de base de datos, utilizando un modelo cliente-servidor.// El servidor será el motor de la base de datos. Atenderá pedidos de altas, modificaciones y consultas por parte de los clientes, y mantendrá y persistirá la información. El mismo es secuencial, por lo que atenderá de a un pedido por vez.// El cliente facilitará la manipulación de la base de datos. A través de una interfaz, el usuario podrá pedir altas y modificaciones de registros al servidor, y realizar consultas. Puede haber más de un cliente ejecutándose al mismo tiempo.// Los registros deben almacenar nombre, dirección y teléfono, utilizando el primero como clave para la base.

2. División del proyecto en procesos

Dado que tanto el servidor como el cliente son secuenciales, solo se distinguen estos dos procesos; que son, además, programas separados.

3. Comunicación entre procesos

Los clientes deben comunicarse con el servidor para enviar los pedidos y consultas, y recibir la respuesta. El servidor debe recibir los pedidos y enviar una respuesta a los clientes. Por lo tanto, se necesita una comunicación que permita enviar y recibir en ambos sentidos.

4. Mecanismos de concurrencia utilizados

4.1. Comunicación cliente-servidor

Para la comunicación entre el cliente y el servidor se utilizará una cola de mensajes. El ID del servidor será un número fijo y determinado en un archivo, y el de cada cliente será su PID. El mismo será utilizado para que cada proceso solo saque de la cola los mensajes que le corresponden, evitando pérdida de mensajes o respuestas inválidas. Esto además permite utilizar la misma cola para todos los clientes, de manera que los recursos consumidos no aumentan según la cantidad de conexiones (salvo la memoria que utiliza cada uno de los mismos).

4.2. Logger

Para el logger se utiliza un lock de escritura, de manera que los distintos procesos no escriban al mismo tiempo en el archivo de log.

5. Diagrama de clases

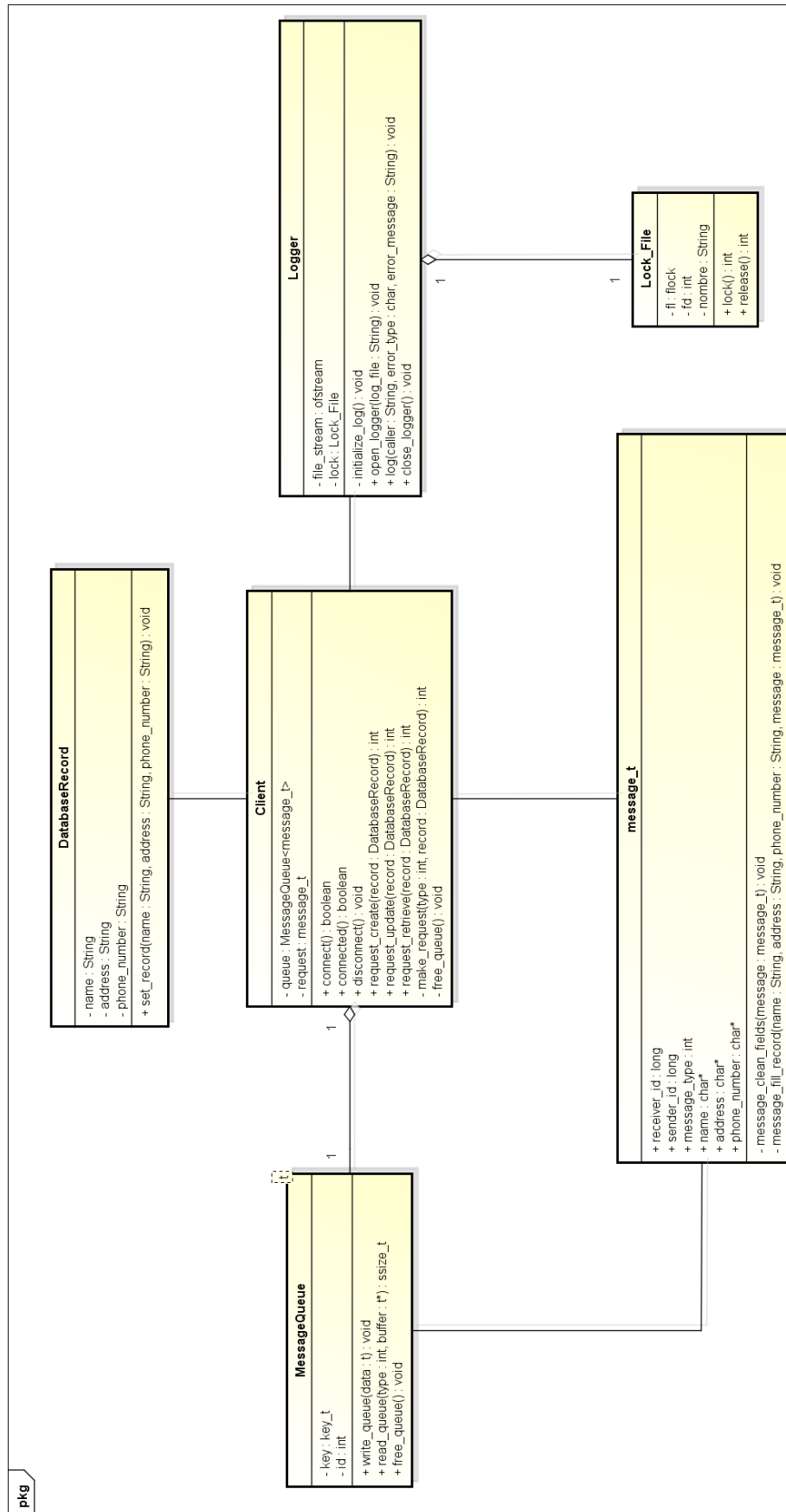


Figura 1: Diagrama de clases del cliente

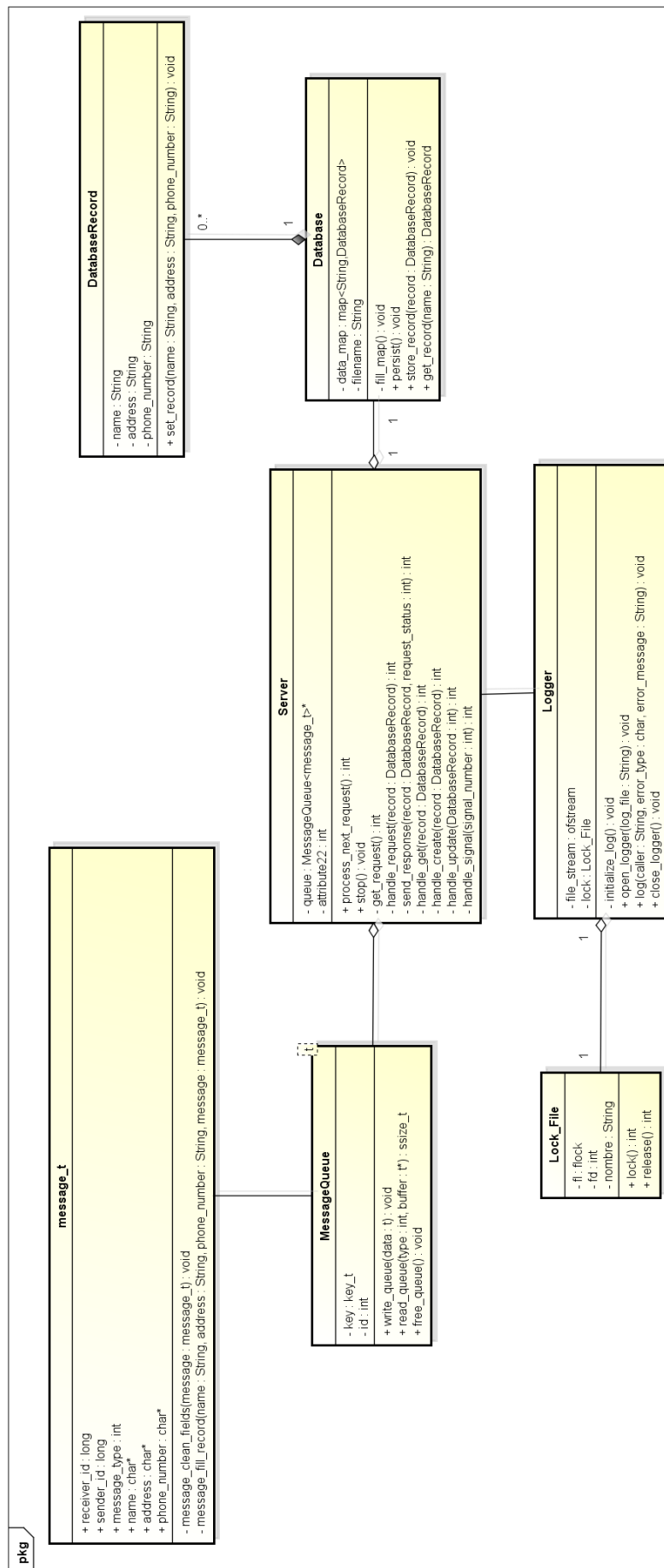
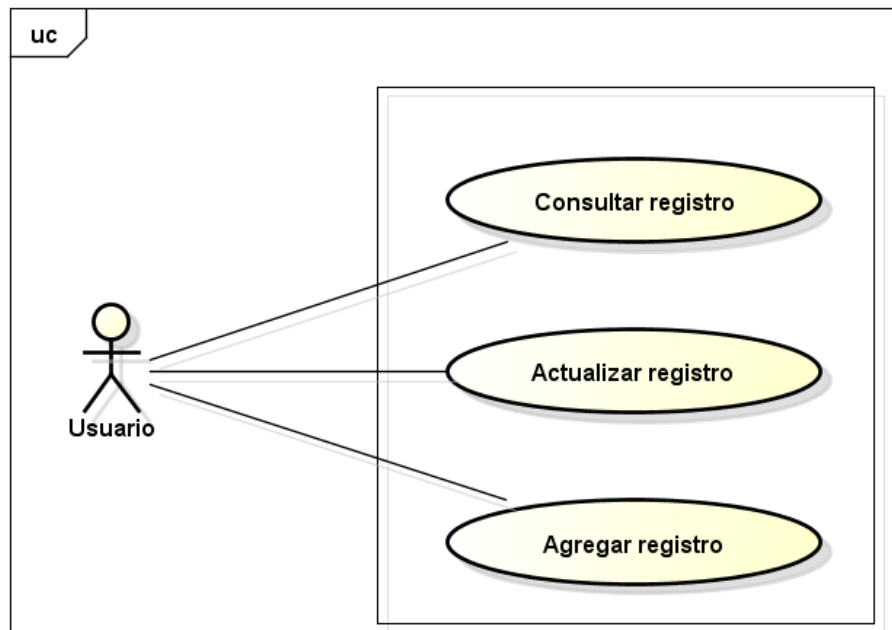


Figura 2: Diagrama de clases del servidor

6. Diagrama de casos de uso



powered by Astah

Figura 3: Diagrama de casos de uso