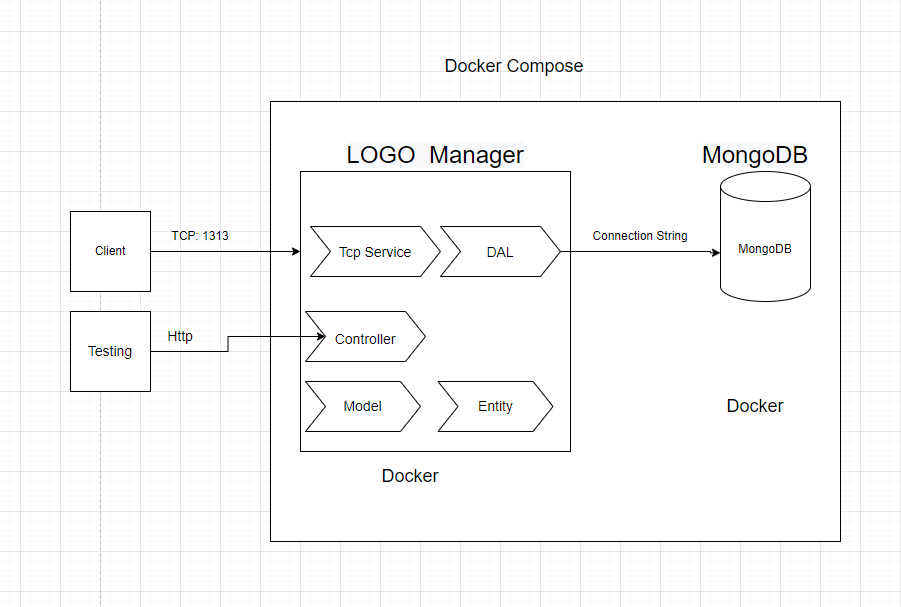
# In Log We Trust

# User Guide

## Summery

## To implement a system that saves log messages from a TCP connection into MongoDB, splitting them by date and client information, follow these steps:

## High level design



## Details Design

Two dockers run in a Docker container:

**3.1 Logo:** The manager docker. **C# .Net Core 6 VS2022**

* + - TCP Service: service that listens to IP address and port.
    - DAL (MongoDB): Connect to MongoDB docker and insert messages into SleekDb.
    - Controller (Just for testing): Can send command to Start to listen to TCP and

insert a message into MongoDB.

* + - Model: Logo Message model that comes from the client side.
    - Entity: Entity in MongoDB Schema.

**3.2 MongoDB**: Database saving all logo messages splitting by date and client information.

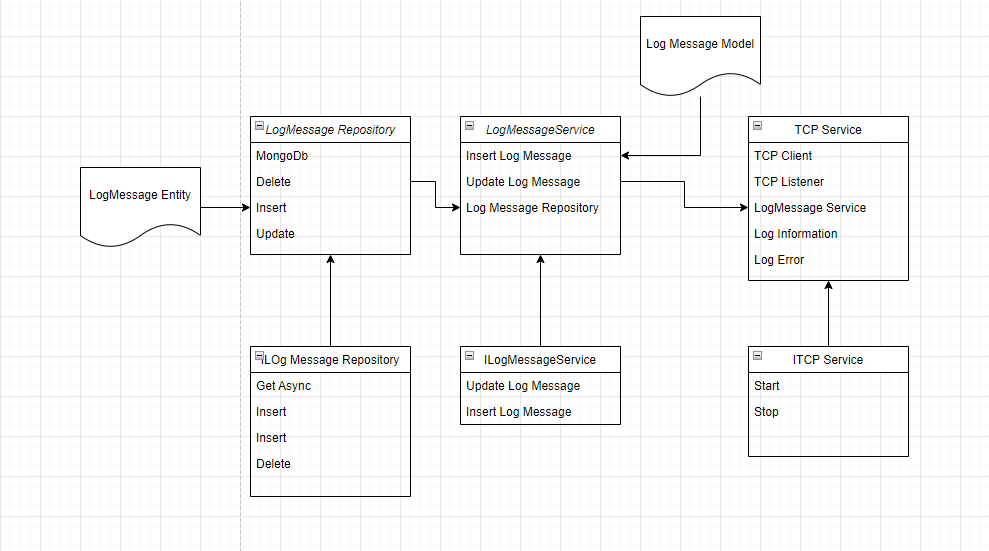
## Flow Diagram

תמונה שמכילה טקסט, צילום מסך, תרשים, מקביל

התיאור נוצר באופן אוטומטי

## Class Diagram

Logo Manager solution is written by VS 2022 C# .Net Core 6.



## Configuration file: appsettings.json / Docker-compose.yml

TcpPort: port number

TcpIp: Ip address