# Proposed software architecture

## Overview

CrypDist is a system which provides mass distributed data storage and synchronized data storage among clients where clients can interact with synchronized local data and mass data via graphical user interface. For this reason, the system is divided into subsystems according to functionality, in order to split the developing complexity of the project. Furthermore, with this decomposition, the independency of functional parts is being tried to be increased so that a subsystem can be changed without changing other subsystems, which increases the extendibility and maintainability of the system.

The main functionalities that CrypDist aims to satisfy are Graphical User Interface, Synchronized Data Storage among clients (BlockChain), local data storage for storing BlockChain data and server side storage where the mass data will be stored.

## Subsystem decomposition

The CrypDist system is decomposed into subsystems according to functionalities. For client software which consist of GUI, BlockChain and Local Data Storage subsystems, 3-Tier Architecture will be followed in order to increase the extendibility and maintainability of the system, as mentioned above. The server software will work independently from client side.

### Graphical User Interface Subsystem

Graphical User Interface Subsystem is “Presentation” layer of 3-Tier Architecture and forms the outside door of the system where the users can interact with the system. The information about BlockChain is visualized in interface and users can perform actions (query the data, download data, update data, store new one) through this interface.

### BlockChain Subsystem

BlockChain Subsystem is “Application” layer of 3-Tier Architecture and it is responsible for synchronizing the summary data among the clients. In initialization step of system, this subsystem read the locally stored data from Local Data Storage Subsystem, in order not to re-download the whole data. For synchronizing the data, this subsystem establishes peer to peer communication between clients. Also, for downloading and updating the distributed mass data, this subsystem has a connection to Mass Data Storage Subsystem. For this connection, Client-Server architecture will be applied and BlockChain Subsystem will form the client side of this connection.

### Local Data Storage Subsystem

Local Data Storage Subsystem is “Data” layer of 3-Tier Architecture and it stores the BlockChain data in local databases which prevents system from re-downloading whole data after first initialization. The local data is updated whenever there is an update on BlockChain, and it is kept synchronized with BlockChain as system is alive.

### Mass Data Storage Subsystem

Mass Data Storage Subsystem is server side of the system and responsible for storing the whole data which cannot be stored locally on clients. This subsystem forms server part of the Client – Server architecture (between BlockChain and Mass Data Storage) which is stated in BlockChain subsystem. This subsystem works independently from client software and it provides download, update and upload links for client software to make the clients able to manipulate the mass data.