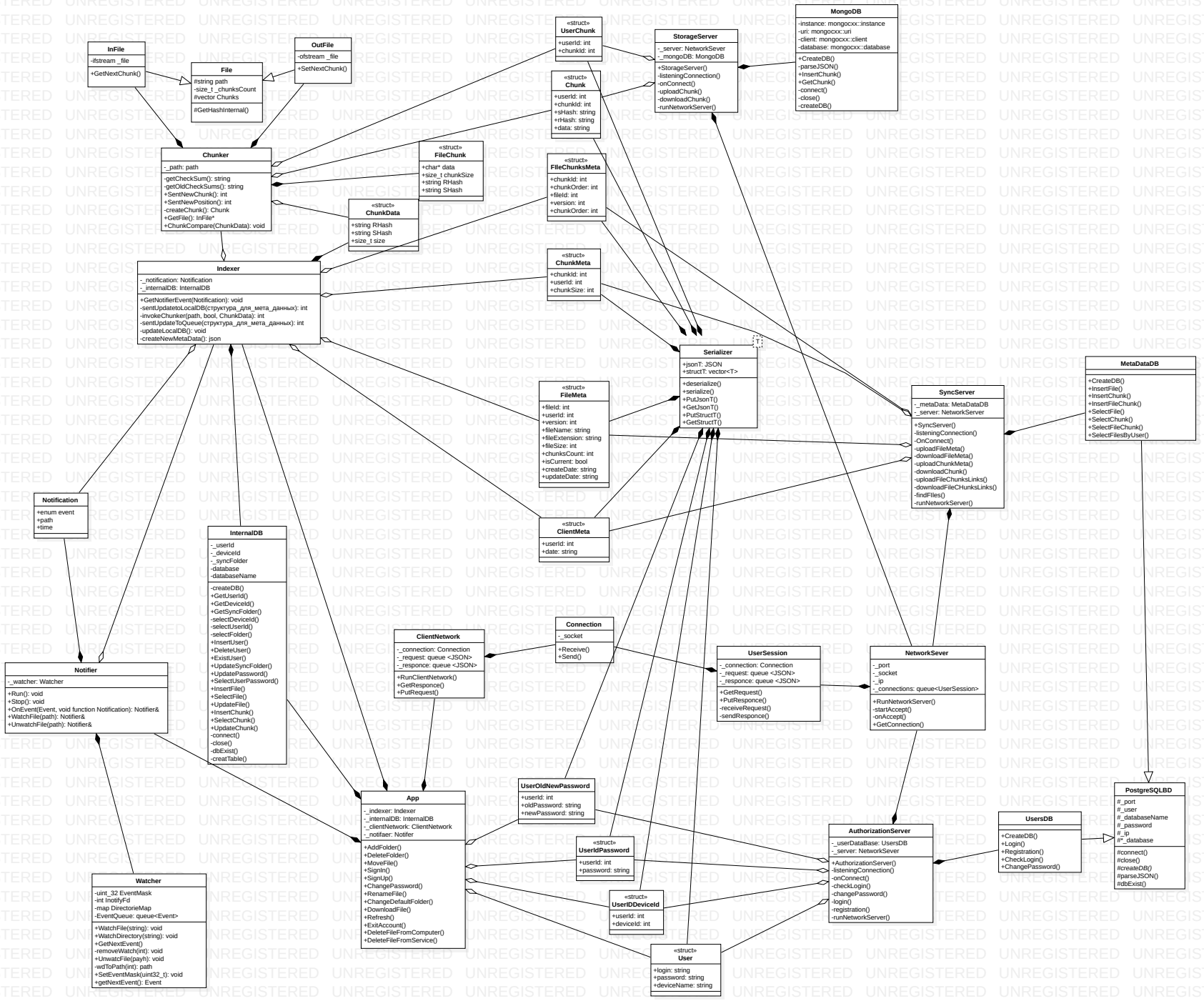
Model.main


InFile
-ifstream _file
+GetNextChunk()
File
#string path
-size_t _chunksCount
#vector Chunks
#GetHashInternal()
OutFile
-ofstream _file
+SetNextChunk()
«struct»
UserChunk
+userId: int
+chunkId: int
StorageServer
-_server: NetworkSever
-_mongoDB: MongoDB
+StorageServer()
-listeningConnection()
-onConnect()
-uploadChunk()
-downloadChunk()
-runNetworkServer()
MongoDB
-instance: mongocxx::instance
-uri: mongocxx::uri
-client: mongocxx::client
-database: mongocxx::database
+CreateDB()
-parseJSON()
+InsertChunk()
+GetChunk()
-connect()
-close()
-createDB()
«struct»
Chunk
+userId: int
+chunkId: int
+sHash: string
+rHash: string
+data: string
Chunker
-_path: path
-getCheckSum(): string
-getOldChecksums(): string
+SentNewChunk(): int
+SentNewPosition(): int
-createChunk(): Chunk
+GetFile(): InFile*
+ChunkCompare(ChunkData): void
«struct»
FileChunk
+char* data
+size_t chunkSize
+string RHash
+string SHash
«struct»
FileChunksMeta
+chunkId: int
+chunkOrder: int
+fileId: int
+version: int
+chunkOrder: int
«struct»
ChunkData
+string RHash
+string SHash
+size_t size
«struct»
ChunkMeta
+chunkId: int
+userId: int
+chunkSize: int
Indexer
-_notification: Notification
-_internalDB: InternalDB
+GetNotifierEvent(Notification): void
-sentUpdateteLocalDB(структура_для_мета_данных): int
-sentInvokeChunker(path, bool, ChunkData): int
-sentUpdateToQueue(структура_для_мета_данных): int
-updateLocalDB(): void
-createNewMetaData(): json
Serializer
T
+jsonT: JSON
+structT: vector<T>
+deserialize()
+serialize()
+PutJsonT()
+GetJsonT()
+PutStructT()
+GetStructT()
MetaDataDB
+CreateDB()
+InsertFile()
+InsertChunk()
+InsertFileChunk()
+SelectFile()
+SelectChunk()
+SelectFileChunk()
+SelectFilesByUser()
«struct»
FileMeta
+fileId: int
+userId: int
+version: int
+fileName: string
+fileExtension: string
+fileSize: int
+chunksCount: int
+isCurrent: bool
+createDate: string
+updateDate: string
SyncServer
-_metaData: MetaDataDB
-_server: NetworkServer
+SyncServer()
-listeningConnection()
-OnConnect()
-uploadFileMeta()
-downloadFileMeta()
-uploadChunkMeta()
-downloadChunk()
-uploadFileChunksLinks()
-downloadFileChunksLinks()
-findFiles()
-runNetworkServer()
Notification
+enum event
+path
+time
«struct»
ClientMeta
+userId: int
+date: string
InternalDB
-_userId
-_deviceId
-_syncFolder
-database
-databaseName
-createDB()
+GetUserId()
+GetDeviceId()
+GetSyncFolder()
-selectDeviceId()
-selectUserId()
-selectFolder()
+InsertUser()
+DeleteUser()
+ExistUser()
+UpdateSyncFolder()
+UpdatePassword()
+SelectUserPassword()
+InsertFile()
-selectFile()
+UpdateFile()
+InsertChunk()
+SelectChunk()
+UpdateChunk()
-connect()
-close()
-dbExist()
-creatTable()
Connection
-_socket
+Receive()
+Send()
ClientNetwork
-_connection: Connection
-_request: queue <JSON>
-_responce: queue <JSON>
+RunClientNetwork()
+GetResponce()
+PutRequest()
UserSession
-_connection: Connection
-_request: queue <JSON>
-_responce: queue <JSON>
+GetRequest()
+PutResponce()
-receiveRequest()
-sendResponce()
NetworkSever
-_port
-_socket
-_ip
-_connections: queue<UserSession>
+RunNetworkServer()
-startAccept()
-onAccept()
+GetConnection()
Notifier
-_watcher: Watcher
+Run(): void
+Stop(): void
+OnEvent(Event, void function Notification): Notifier&
+WatchFile(path): Notifier&
+UnwatchFile(path): Notifier&
UserOldNewPassword
+userId: int
+oldPassword: string
+newPassword: string
App
-_indexer: Indexer
-_internalDB: InternalDB
-_clientNetwork: ClientNetwork
-_notifiaer: Notifer
+AddFolder()
+DeleteFolder()
+MoveFile()
+SignIn()
+SignUp()
+ChangePassword()
+RenameFile()
+ChangeDefaultFolder()
+DownloadFile()
+Refresh()
+ExitAccount()
+DeleteFileFromComputer()
+DeleteFileFromService()
«struct»
UserIdPassword
+userId: int
+password: string
«struct»
UserIDDeviceId
+userId: int
+deviceId: int
«struct»
User
+login: string
+password: string
+deviceName: string
AuthorizationServer
-_userDataBase: UsersDB
-_server: NetworkSever
+AuthorizationServer()
-listeningConnection()
-onConnect()
-checkLogin()
-changePassword()
-login()
-registration()
-runNetworkServer()
UsersDB
+CreateDB()
+Login()
+Registration()
+CheckLogin()
+ChangePassword()
PostgreSQLBD
#_port
#_user
#_databaseName
#_password
#_ip
#*_database
#connect()
#close()
#creaseDB()
#parseJSON()
#dbExist()
Watcher
-uint_32 EventMask
-int InotifyFd
-map DirectorieMap
-EventQueue: queue<Event>
+WatchFile(string): void
+WatchDirectory(string): void
+GetNextEvent()
-removeWatch(int): void
+UnwatcFile(path): void
-wdToPath(int): path
+SetEventMask(uint32_t): void
+getNextEvent(): Event