DatabaseManager manages the connection to the datastore It optimizes the connection to the datastore using caching, and other capabilities to be added over time. It provides a local observer design pattern to let the rest of the application know when there are updates. Ideally, also eventually implement observer design pattern against the server backend, so polling isn't needed.

NOTE: The local in-memory cache will allow the application to refresh as much as it want's, and not worry about db impact, or user experience impact.

NOTE: Since DatabaseManager is doing caching, when a method is called that returns a reference to data that is cached, the cache (maybe in a dictionary) should be searched so that if that object has already been created and a reference provided before, that same reference is provided again.

DatabaseManager Singleton · QMap<QString, TvShow> tvShowMap QMap<QString, User> userMap - IsonConnection connection + DatabaseManager() + DatabaseManager(QString jsonPath) + const TvShow inspectTvShow(QString name) + const TvShow inspectTvShowById(int tvShowId) + const User inspectUser(QString username) + const User inspectUserByEmail(QString email) + void addUser(User user) + void removeUser(QString username) + bool usernameExists(QString username) + bool emailExists(QString email) + QList<QString> getListOfAllTvShows() + QList<QString> getListOfAllUsers() + QList<QString> getListOfCachedTvShows() + QList<QString> getListOfCachedUsers() + QString getTvShowNameById(int showId) + void addEpisodeReview(EpisodeIdentifier episodeIdentifier, Review review) + void addEpisodeComment(EpisodeIdentifier episodeIdentifier, Comment comment) + void addWatchedEpisode(EpisodeIdentifier episodeIdentifier, QString username) + void removeWatchedEpisode(EpisodeIdentifier episodeIdentifier, QString username) + QList<QString> getListOfWatchedTvShowNamesForUser(QString username) + void emptyCache() > void notify() DatabaseManager(DatabaseManager const&)

0..1

Add data to the datastore Modify data in the data store iConnection + virtual TvShow getTvShow(QString name) = 0 + virtual void addTvShow(TvShow tvShow) = 0 + virtual User getUser(QString username) = 0 + virtual QString getUserNameByEmail(QString email) = 0 + virtual QString getTvShowNameById(int id) = 0 + virtual void addUser(User user) = 0 + virtual void removeUser(QString username) = 0 + virtual bool usernameExists(QString username) = 0 + virtual bool emailExists(QString email) = 0 + virtual QList<QString> getListOfAllTvShows() = 0 + virtual QList<QString> getListOfAllUsers() = 0 + virtual void addEpisodeReview(EpisodeIdentifier episodeIdentifier, Review review) = 0 + virtual void addEpisodeComment(EpisodeIdentifier episodeIdentifier, Comment comment) = 0 + virtual void addWatchedEpisode(EpisodeIdentifier episodeIdentifier, QString username) = 0 CsvConnection + virtual void removeWatchedEpisode(EpisodeIdentifier episodeIdentifier, QString username) = 0 -pathToCsvFolder NotImplemented() One Connection at a time only. IsonConnection + QJsonObject json const QString JSON_USER_ARRAY_NAME const QString JSON_TVSHOW_ARRAY_NAME OString pathToJson + JsonConnection() + JsonConnection(QString pathToJson) + OString getPathToJson() + void setPathToJson(QString newPathToJson) + TvShow getTvShow(QString name) + void addTvShow(TvShow tvShow) + User getUser(QString username) + OString getUserNameByEmail(OString email) + QString getTvShowNameById(int id) + void addUser(User user) + void removeUser(QString username) + bool usernameExists(QString username) + bool emailExists(QString email) MySqlConnection + QList<QString> getListOfAllTvShows() + QList<QString> getListOfAllUsers() -OString host + void addEpisodeReview(EpisodeIdentifier episodeIdentifier, Review review) -int port + void addEpisodeComment(EpisodeIdentifier episodeIdentifier, Comment comment) -QString username + void addWatchedEpisode(EpisodeIdentifier episodeIdentifier, QString username) -OString password + void removeWatchedEpisode(EpisodeIdentifier episodeIdentifier, QString username) -QString database + void load1son() + void saveJson() NotImplemented() QJsonArray getTopLevelJsonArray(QString jsonArrayName) - QMap<int, Season> getSeasons(QJsonArray jsonSeasons) Season getSeason(QJsonObject jsonSeason) QMap<int, Episode> getEpisodes(QJsonArray jsonEpisodes) Episode getEpisode(QJsonObject jsonEpisode) QMap<QString, Review> getReviews(QJsonValue jsonReviews) QList < Comment > getComments(QJsonValue jsonComments) QJsonObject tvShowToJsonObject(TvShow tvShow) QJsonArray seasonsToJsonArray(QVector<Season> seasons) QJsonValue seasonToJsonValue(Season season) QJsonArray episodesToJsonArray(QVector<Episode> episodes)

QJsonValue episodeToJsonValue(Episode episode) OJsonArray reviewsToJsonArray(OList<Review> reviews) - QJsonValue reviewToJsonValue(Review review)

· QJsonArray commentsToJsonArray(QList<Comment> comments) - QJsonValue commentToJsonValue(Comment comment) QJsonObject userToJsonObject(User user)

QJsonArray watchedEpisodesToJsonArray(QList<EpisodeIdentifier> watchedEpisodes) QList<EpisodeIdentifier> getWatchedEpisodes(QJsonArray jsonWatchedEpisodes)

Connection defines the methods the application can use to:

Get data from the datastore