

Services -database : DatabaseClient -storage : StorageClient +getDatabaseClient() : DatabaseClient +getStorageClient(): StorageClient <<Interface>> DatabaseClient +getUser() : Future<MealerUser> +uploadFile(byte[] bytes, String remotePath, StorageProgressCallback callback) : Future<Boolean> +getUser(DatabaseCompletionCallback callback) : Future<MealerUser> +getUser(String id) : Future<MealerUser> +getUser(String id, DatabaseCompletionCallback callback) : Future<MealerUser> +getUsers(DatabaseFilterCallback filter) : Future<List<MealerUser>> +getUserMenu(): Future<MealerMenu> +getUserMenu(DatabaseCompletionCallback callback): Future<MealerMenu> DatabaseCompletionCallback<T> +getUserMenu(String id): Future<MealerMenu> +onComplete(T object) : void +getUserMenu(String id, DatabaseCompletionCallback callback): Future<MealerMenu> +getMenus(DatabaseFilterCallback filter): Future<List<MealerUser>> DatabaseFilterCallback +getMenu(String id) : Future<MealerMenu> +applyFilter(CollectionReference reference) : Query +getMenu(String id, DatabaseCompletionCallback callback) : Future<MealerMenu> +getRecipes(DatabaseFilterCallback filter): Future<List<MealerRecipe>> DatabaseSetCallback +getRecipe(String id) : Future<MealerRecipe> +onComplete(Boolean object) : void +getRecipe(String id, DatabaseCompletionCallback callback) : Future<MealerRecipe> +updateRecipe(MealerRecipe recipe): Future<Boolean> StorageProgressCallback +updateRecipe(MealerRecipe recipe, DatabaseSetCallback callback): Future<Boolean> +onProgress(long processed, long total) : void +updateMenu(MealerMenu menu) : Future<Boolean> +updateMenu(MealerMenu menu, DatabaseSetCallback callback) : Future<Boolean> +updateUser(MealerUser user) : Future<Boolean> +updateUser(MealerUser user, DatabaseSetCallback callback) : Future<Boolean> +deleteRecipe(String id) : Future<Boolean> +deleteRecipe(String id, DatabaseSetCallback callback) : Future<Boolean> +userInfoRequired(): Future<Boolean> +userInfoRequired(DatabaseCompletionCallback callback) : Future<Boolean> +listenForModel(Activity activity, String collectionId, String documentId, Class<T> clazz, DatabaseCompletionCallback<T> callback): void FirebaseDatabaseClient

-getModel(String collectionId, String documentId, Class<T> clazz, DatabaseCompletionCallback<T> callback): Future<T>)

-saveModel(String collectionId, String documentId, Object objToSave, DatabaseSetCallback callback) : Future<Boolean>)

-getModels(String collectionId,Class<T> clazz, int limit, DatabaseFilterCallback filter) : Future<List<T>>)

-deleteModel(String collectionId, String documentId, DatabaseSetCallback callback) : Future<Boolean>)

<<Singleton>>

<<Interface>>

StorageClient

FirebaseStorageClient

<<Interface>>

<<Interface>>

<<Interface>>