

Registrierung

+BenutzerExistiert(User&):bool
+CheckIfEmailExists(QString&):bool
+RegistriereBenutzer(User&):bool
+ValidiereEmail(QString):bool

Ernährungsplan

-id_: int
-kaloriengehalt_: int
+start_datum: QString
+end_datum: QString
+beschreibung: QString
+mahlzeiten:
std::vector<Mahlzeit*>

+Erneahrungsplan()
+~~Erneahrungsplan()
+ GetID(): int
+ GetKaloriengehalt(): int
-BerechneKaloriengehalt():int

Mahlzeit

+id_: int
+user_id: int
+kaloriengehalt_: int
+erneahrungsplan_id: int
+uhrzeit_: QString
+datum_: QString
+speisepositionen:
std::vector<Speiseposition*>

+GetKaloriengehalt(): int
+Mahlzeit()
+~Mahlzeit()

Speiseposition

+beschreibung_QString
+kaloriengehalt:int
+menge_:int
+speise_id:int

+Speiseposition()
+~Speiseposition()

Statistik

+abgeschlossene_E_plaene:int
+abgeschlossene_T_einheiten:int
+beschreibung:QString
+end_Datum:QString
+kalorienaufnahme:int
+kalorienverbrauch:int
+start_datum_:QString

+Statistik()
+~Statistik()

Trainingsplan

+beschreibung_:QString
+end_datum_:QString
-id_:int
+start_datum:QString
+trainingseinheiten_:
std::vector<Trainingseinheit*>
+user_id_:int

+ErstelleTrainingsplan():void
+GetID(): int
+GetKalorienverbrauch: int
+Trainingsplan()
+Traningsplan(int,int,QString,
QString,QString)
+~Trainingsplan()

Trainingseinheit

+datum_:QString
+durchgefuehrt:bool
+end_zeit_:QString
+id_:int
+kalorienverbrauch:int
+start_zeit:QString
+trainingsplan_id_:int
+uebungspositionen:
std::vector<Uebungsposition*>
+user_id_:int

+Trainingseinheit()
+Trainingseinheit(int,int,int,bool,
QString, QString, QString)
+~Trainingseinheit()

Uebungsposition

+beschreibung_:QString
+kalorienverbrauch_:int
-id_:int
+menge_:int
+trainingseinheiten_id:int
+uebungs_id_:int

+KalorienverbrauchBerechnen():int
+Uebungsposition()
+Uebungsposition(int,int,int,int,QString)
+~Uebungsposition

DBHandler

-db_:QSqlDatabase

+AddMahlzeit(Mahlzeit&):int
+AddSpeiseposition(Speiseposition&,int):int
+AddTrainingsienheit(Trainingseinheit&):int
+AddTrainingsplan(Trainingsplan&):int
+AddUebungsposition(Uebungsposition&):bool
+CustomQuery(QString):bool
+DBHandler()
+~DBHandler()
+GetErnearhungsPlaeneVonBis(int,QString,QString): std::vector<Ernaerungsplan*>
+GetInstance():DBHandler*
+GetMahlzeitenVonBis(int,QString,QString):std::vector<Mahlzeit*>
+GetRandomUebungen(int,QString): std::vector<Uebungspositionen*>
+GetTrainingseinheit(int): std::vector<Trainingseinheit*>
+GetTrainingsplan(): Trainingsplan*
GetUebungsposition(int): std::vector<Uebungsposition*>
+UpdateUebungsposition(int,int): bool

Profil

-alter_:int
+geb_datum:QString
+geschlecht_:int
-grundumsatz_:int
+height_:int
+kalorienaufnahme_:int
+name_:QString
+prefaerenz_:QString
+vorname_:QString
+weight_:int

+AddProfile(Profil&, int): bool
-AlterBerechnen(QString): int
+GetAlter():int
+GetGrundumsatz():int
+GetProfil(int):Profil*
+GetProfil(User&):Profil*
-GrundumsatzBerechnen(int,int,int,int):int
+Profil()
+~Profil()
+UpdateProfil(Profil&,int):bool

User

+email_:QString
-id_:int
+password_:QString

+AddUser(User&):bool
+DeleteUser(int):bool
+GetId():int
+GetUser(int):User*
+GetUser(QString):User*
+GetUser(QString,QString):User*
+Setid(int):void
+UpdateUser(User&):bool
+User(QString,QString)
+User(QString,QString,int)
+~User()

Login

-logged_in:user_:User*

+ GetInstance(): Login*
+GetLoggedInUser():User*
+LoginUser(User&):bool

+mahlzeiten

0..*

+speisepositionen

0..*

+trainingseinheiten0..*

+uebungsposition0..*

-logged_in_user