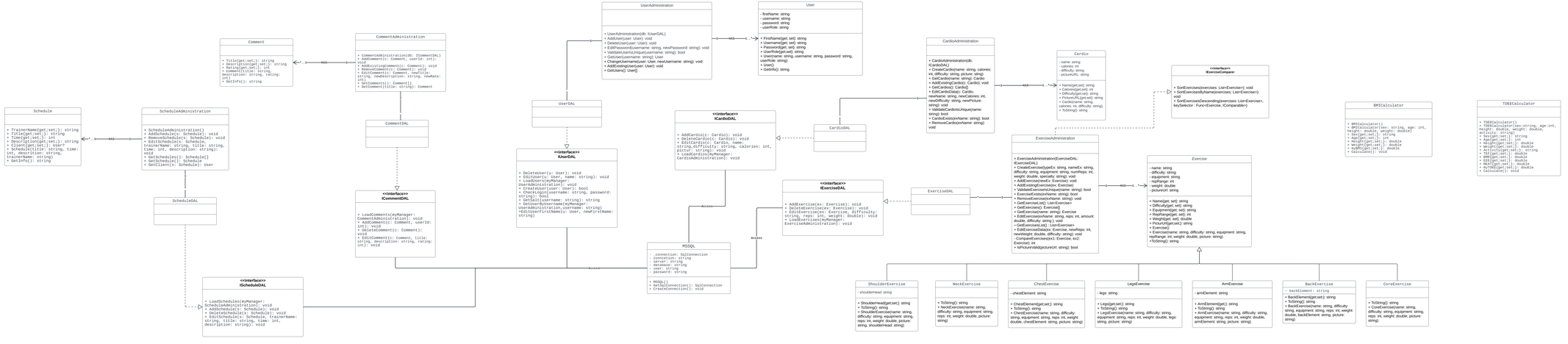
Schedule
+ TrainerName(get;set;): string
+ Title(get;set;): string
+ Time(get;set;): int
+ Description(get;set;): string
+ Client(get;set;): user?
+ AddSchedule(trainer: string, time: int, description: string, trainerName: string)
+ GetInfo(): string
1..*
HAS
1
ScheduleAdministration
+ ScheduleAdministration()
+ AddSchedule(s: Schedule): void
+ DeleteSchedule(trainerName): void
+ EditSchedule(s: Schedule, trainerName: string, title: string, time: int, description: string): void
+ Load
+ GetSchedules(): Schedule[]
+ GetSchedule(): Schedule
+ GetClient(s: Schedule): User
ScheduleDAL

Comment
+ Title(get;set;): string
+ Description(get;set;): string
+ Rating(get;set;): int
+ Comment(title: string, description: string, rating: int)
+ GetInfo(): string
1..*
HAS
1
CommentAdministration
+ CommentAdministration(db: ICommentDAL)
+ AddComment(c: Comment, userId: int): void
+ AddExistingComment(c: Comment): void
+ RemoveComment(c: Comment): void
+ EditComment(c: Comment, newTitle: string, newDescription: string, newRate: int)
+ GetComments(): Comment[]
+ GetComment(title: string): Comment
CommentDAL

UserAdministration
+ UserAdministration(db: IUserDAL)
+ AddUser(user: User): void
+ DeleteUser(user: User): void
+ EditPassword(username: string, newPassword: string): void
+ ValidateUsersUnique(username: string): bool
+ GetUser(username: string): User
+ ChangeUsername(user: User, newUsename: string): void
+ AddExistingUser(user: User): void
+ GetUsers(): User[]
1..*
HAS
1..*
User
- firstName: string
- username: string
- password: string
- userRole: string
+ FirstName(get; set): string
+ Username(get; set): string
+ Password(get; set): string
+ UserRole(get;set): string
+ User(name: string, username: string, password: string, userRole: string)
+ User()
+ GetInfo(): string
UserDAL

<<interface>>
ICardioDAL
+ AddCardio(c: Cardio): void
+ DeleteCardio(c: Cardio): void
+ EditCardio(c: Cardio, name: string,difficulty: string, calories: int, pictur: string): void
+ LoadCardios(myManager: CardioAdministration): void
Access
MSSQL
- _connection: SqlConnection
- connection: string
- server: string
- database: string
- user: string
- password: string
+ MSSQL()
+ GetSqlConnection(): SqlConnection
+ CreateConnection(): void
Access
CardioAdministration
+ CardioAdministration(db: ICardioDAL)
+ CreateCardio(name: string, calories: int, difficulty: string, picture: string)
+ GetCardio(name: string): Cardio
+ AddExistingCardio(c: Cardio): void
+ GetCardios(): Cardio[]
+ EditCardio(name: Cardio, newName: string, newCalories: int, newDifficulty: string, newPicture: string): void
+ ValidateCardiosUnique(name: string): bool
+ CardioExists(exName: string): bool
+ RemoveCardio(exName: string): void
1..*
HAS
1..*
Cardio
- name: string
- calories: int
- difficulty: string
- pictureURL: string
+ Name(get;set): string
+ Calories(get;set): int
+ Difficulty(get;set): string
+ PictureURL(get;set): string
+ Cardio(name: string, calories: int, difficulty: string)
+ ToString(): string
CardioDAL

<<interface>>
IExerciseDAL
+ AddExercise(ex: Exercise): void
+ DeleteExercise(ex: Exercise): void
+ EditExercise(ex: Exercise, difficulty: string, reps: int, weight: double): void
+ LoadExercises(myManager: ExerciseAdministration): void
Access
ExerciseDAL
ExerciseAdministration
+ ExerciseAdministration(IExerciseDAL: IExerciseDAL)
+ CreateExercise(typeEx: string, nameEx: string, difficulty: string, equipment: string, numReps: int, weight: double, specialty: string): void
+ AddExercise(newEx: Exercise): void
+ AddExistingExercise(ex: Exercise)
+ ExerciseExists(exercise: Exercise, name): bool
+ ExerciseExists(exName: string): bool
+ RemoveExercise(exName: string): void
+ GetExercise(Last): List<Exercise>
+ GetExercises(): Exercise[]
+ GetExercise(name: string): Exercise
+ EditExercise(exName: string, reps: int, amount: int, weight: double): void
+ GetExercises(Last): List<Exercise>
+ EditExercise(Dataprx: Exercise, newReps: int, newWeight: double, difficulty: string): void
- CompareExercises(ex1: Exerose, ex2: Exercise): int
+ IsPictureValid(pictureUrl: string): bool
1..*
HAS
1..*
Exercise
- name: string
- difficulty: string
- equipment: string
- repRange: int
- weight: double
- pictureUrl: string
+ Name(get; set): string
+ Difficulty(get; set): string
+ Equipment(get; set): string
+ RepRange(get; set): int
+ Weight(get; set): double
+ PictureUrl(get;set): string
+ Exercise()
+ Exercise(name: string, difficulty: string, equipment: string, repRange: int, weight: double, picture: string)
+ ToString(): string