llert

Project Report
November 30, 2018

Luke Duhe
JJ Juarez
Drake Lambert
Kevin Phan
Sam Miller
Tristan Miller
Timothy Ratliff
Steven Vondenstein
William Woodfin

Contents

1 Database Design

1.1 Introduction

1.2 ER Diagram

1.3 Identified Constraints

- Every application user must have both a username and a password
- Usernames must be unique
- Every board must have a title and must be associated with an existing owner
- Every comment must have an owner
- Every comment must have a time and text
- Every comment must be associated with an existing task
- Every list must have a title and must be associated with an existing board
- Every task must be associated with an existing list
- Every entity in every table has a unique id

1.4 Assumptions about the domain

- Users should only have access to the boards they are members of

1.5 database design process

1.6 our tables

1.6.1 Functional Dependencies

1.6.2 Primary and Foreign Keys

2 Database Implementation

2.1 Create Table Statements

```sql
create table ApplicationUser(
  Id char(32),
  Username varchar(100) not null,
  Passwordhash varchar(100) not null,
  primary key ( Id )
);

```

Listing 1: ApplicationUser Table

```sql
      create table Board(
        Id char(32),
        Title varchar(100) not null,
        OwnerId char(32),
        primary key ( id ),
        foreign key ( OwnerId ) references ApplicationUser( Id )
      );

```

Listing 2: Board Table

```sql
      create table BoardMember(
        BoardId char(32),
        MemberId char(32),
        foreign key ( BoardId ) references Board( Id ),
        foreign key ( MemberId ) references ApplicationUser( Id )
      );

```

Listing 3: BoardMember Table

```sql
      create table List(
        Id char(32),
        Title varchar(100) not null,
        BoardPosition int not null,
        BoardId char(32),
        primary key ( id ),
        foreign key ( BoardId ) references Board( Id )
      );

```

Listing 4: List Table

```sql
      create table Task(
        Id char(32),
        Title varchar(100) not null,
        Descriptor varchar(500) not null,
        DueDate datetimeoffset,
        ListId char(32),
        primary key ( id ),
        foreign key ( ListId ) references List( Id )
      )

```

Listing 5: Task Table

```sql
      create table TaskAssignee(
        TaskId char(32),
        AssigneeId char(32),
        foreign key ( TaskId ) references Task( Id ),
        foreign key ( AssigneeId ) references ApplicationUser( Id )
      )

```

Listing 6: TaskAssignee Table

```
create table Comment(
  Id char(32),
  TimeCreated datetimeoffset not null,
  MessageText varchar(100) not null,
  TaskId char(32),
  OwnerId char(32),
  primary key ( id ),
  foreign key ( TaskId ) references Task( Id ),
  foreign key ( OwnerId ) references ApplicationUser( Id )
)

```

Listing 7: Comment Table

2.2 Insert Statements

2.3 Data manipulation statements

2.3.1 Select statements

2.3.2 Other Statements

2.3.3 Update statements