"Ollert

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 Ollert's tables

1.6.1 Functional Dependencies

1.6.2 Primary Keys

- ApplicationUser

  - ApplicationUser.Id

- Board

  - Board.Id

- BoardMember

  - none

- List

  - List.Id

- Task

  - Task.Id

- TaskAssignee

  - none

- Comment

  - Comment.Id

1.6.3 Foreign Keys

- Board

  - ApplicationUser.Id

- BoardMember

  - Board.Id
  - ApplicationUser.Id

- List

  - Board.Id

- Task

  - List.Id

- TaskAssignee

  - Task.Id
  - ApplicationUser.Id

- Comment

  - Task.Id
  - ApplicationUser.Id

2 Database Implementation

2.1 Create Table Statements

```sql
1    create table ApplicationUser(
2      Id char(32),
3      Username varchar(100) not null,
4      Passwordhash varchar(100) not null,
5      primary key ( Id )
6    );
7
```

Listing 1: ApplicationUser Table Creation Statement

```sql
1    create table Board(
2      Id char(32),
3      Title varchar(100) not null,
4      OwnerId char(32),
5      primary key ( id ),
6      foreign key ( OwnerId ) references ApplicationUser( Id )
7    );
8
```

Listing 2: Board Table Creation Statement

```sql
1    create table BoardMember(
2      BoardId char(32),
3      MemberId char(32),
4      foreign key ( BoardId ) references Board( Id ),
5      foreign key ( MemberId ) references ApplicationUser( Id )
6    );
7
```

Listing 3: BoardMember Table Creation Statement

```sql
1    create table List(
2      Id char(32),
3      Title varchar(100) not null,
4      BoardPosition int not null,
5      BoardId char(32),
6      primary key ( id ),
7      foreign key ( BoardId ) references Board( Id )
8    );
9
```

Listing 4: List Table Creation Statement

```sql
1    create table Task(
2      Id char(32),
3      Title varchar(100) not null,
4      Descriptor varchar(500) not null,
5      DueDate datetimeoffset,
6      ListId char(32),
7      primary key ( id ),
```

```
8          foreign  key  (  ListId  )  references  List (  Id  )
9        )
10
```

Listing 5: Task Table Creation Statement

```
1        create  table  TaskAssignee (
2          TaskId  char (32) ,
3          AssigneeId  char (32) ,
4          foreign  key  (  TaskId  )  references  Task (  Id  ) ,
5          foreign  key  (  AssigneeId  )  references  ApplicationUser (  Id  )
6        )
7
```

Listing 6: TaskAssignee Table Creation Statement

```
1        create  table  Comment (
2          Id  char (32) ,
3          TimeCreated  datetimeoffset  not  null ,
4          MessageText  varchar (100)  not  null ,
5          TaskId  char (32) ,
6          OwnerId  char (32) ,
7          primary  key  (  id  ) ,
8          foreign  key  (  TaskId  )  references  Task (  Id  ) ,
9          foreign  key  (  OwnerId  )  references  ApplicationUser (  Id  )
10        )
11
```

Listing 7: Comment Table Creation Statement

2.2   Insert Statements

```
1    insert  into  ApplicationUser  values  (" 19843875 −6077−49" ,  " Walter
       Rogers" ,  " 168
       E5F6A717237FB2232A8AFE2DAAE3F8D582C5D4CC0EAA268F05F420F1EC421" )
       ;
2
3    insert  into  ApplicationUser  values  (" 372 a9ad5 −4952−44" ,  " Jean
       Bryant" ,  "
       DAB12D7BB613EAC0304D9917738729FB37B60EBB1FB59FC9493ED64733CCE3BA
       " ) ;
```

Listing 8: ApplicationUser Insert Statements

```
1    insert  into  Board  values  (" bbdd7100−cd10 −41" ,  " Sports  Forum
       Mobile  App" ,  " 19843875 −6077−49" ) ;
2
3    insert  into  Board  values  (" 62376bd0 −6ecc −4f" ,  " Untitled
       Platformer  Game" ,  " 372 a9ad5 −4952−44" ) ;
```

Listing 9: Board Insert Statements

```
1    insert  into  BoardMember  values  (" fdd8f8a8 −0d53−4f" ,  "
       19843875 −6077−49" ) ;
2
```

```
3  insert into BoardMember values ("bbdd7100−cd10−41", "372a9ad5
     −4952−44");
```
Listing 10: BoardMember Insert Statements

```
1  insert into List values ("64251244−40f5−45","Admin Website", 2, "
     bbdd7100−cd10−41");
2
3  insert into List values ("a583557a−3d83−4a","Art Design", 0, "
     62376bd0−6ecc−4f");
```
Listing 11: List Insert Statements

```
1  insert into Task values ("e01479d6−6019−4a", "Database Design", "
     Design a robust database schema for storing all the data in our
      app.", "20180120 09:00:00 +10:00", "0e72d679−da23−41");
2
3  insert into Task values ("d46993c0−ce64−46", "Story Design", "
     Write a fun story for the game.", "20180620 09:00:00 +10:00", "
     a583557a−3d83−4a");
```
Listing 12: Task Insert Statements

```
1  insert into TaskAssignee values ("ac1a2b65−d1fc−47","88b03172−135
     a−4b");
2
3  insert into TaskAssignee values ("a9f5ae59−6193−40","d263667b−
     ea46−4b");
```
Listing 13: TaskAssignee Insert Statements

```
1  insert into Comment values("c088cfa0−15e2−4f", "20180110 08:54:00
      +10:00", "I'm probably going to need some help with this.", "
     e01479d6−6019−4a", "372a9ad5−4952−44");
2
3  insert into Comment values("59646b1b−a084−49", "20180120 08:54:00
      +10:00", "So I'm thinking our game has a mario−like character
     − but green.", "a9f5ae59−6193−40", "d263667b−ea46−4b");
```
Listing 14: Comment Insert Statements

2.3   Data manipulation statements

2.3.1   Select statements

2.3.2   Other Statements

2.3.3   Update statements

3   Index

Listings