

# *"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

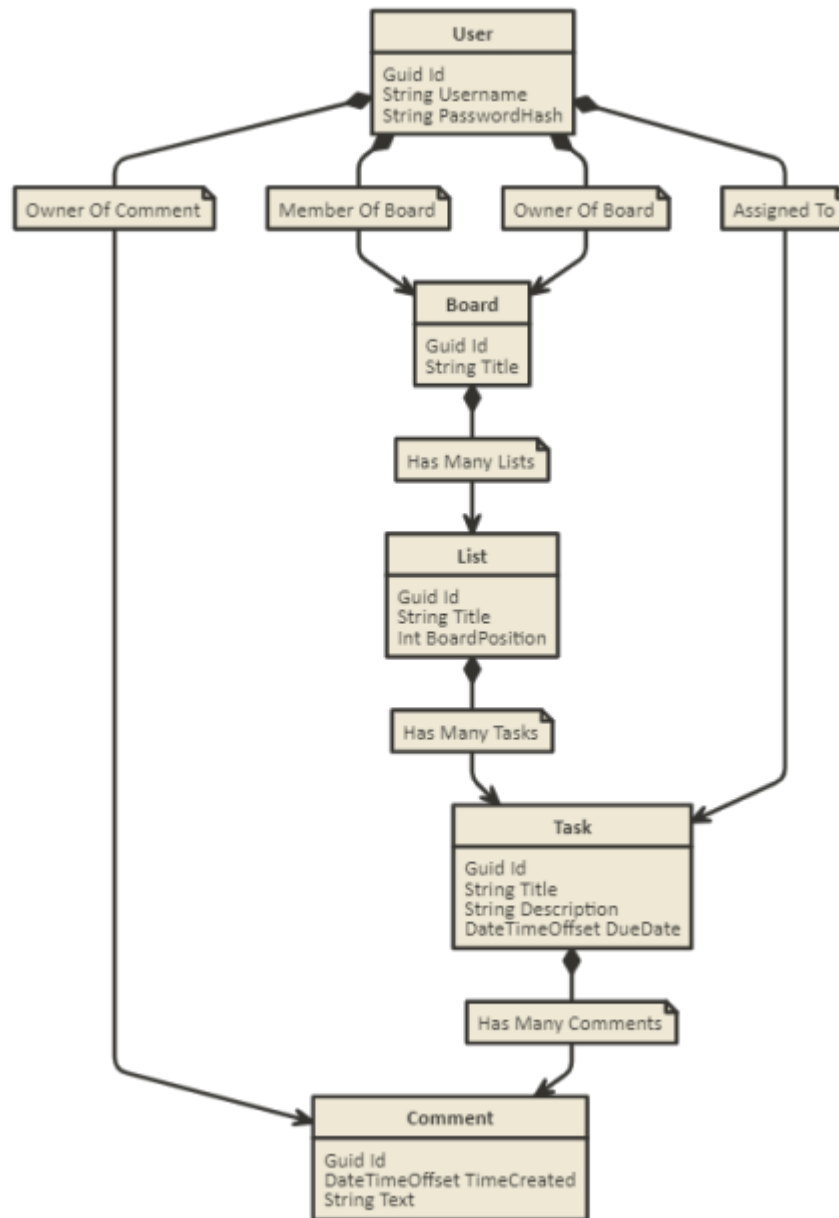
<b>1</b>	<b>Database Design</b>	<b>2</b>
1.1	Introduction . . . . .	2
1.2	ER Diagram . . . . .	3
1.3	Identified Constraints . . . . .	4
1.4	Assumptions About the Domain . . . . .	4
1.5	Database Design Process . . . . .	4
1.6	Ollert's tables . . . . .	4
<b>2</b>	<b>Database Implementation</b>	<b>6</b>
2.1	Create Table Statements . . . . .	6
2.2	Insert Statements . . . . .	7
2.3	Data manipulation statements . . . . .	9
<b>3</b>	<b>Index</b>	<b>12</b>

# 1 Database Design

## 1.1 Introduction

Ollert is the backend database for a Kanban Board application (similar to <https://Trello.com>). It supports multiple users collaborating on one or many boards, through comments and task generation.

## 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

```
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

```
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

```
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

```
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

```
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
2      Rogers",
3      "168
4      E5F6A717237FB2232A8AFE2DAAE3F8D582C5D4CC0EAA268F05F420F1EC421" )
5      ;
6
7  insert into ApplicationUser values ( "372a9ad5-4952-44", "Jean
8      Bryant",
9      "
10     DAB12D7BB613EAC0304D9917738729FB37B60EBB1FB59FC9493ED64733CCE3BA
11     " );

```

Listing 8: ApplicationUser Insert Statements

```

1  insert into Board values ( "bddd7100-cd10-41", "Sports Forum
2      Mobile App", "19843875-6077-49" );
3
4  insert into Board values ( "62376bd0-6ecc-4f", "Untitled
5      Platformer Game", "372a9ad5-4952-44" );

```

Listing 9: Board Insert Statements

```

1  insert into BoardMember values ( "fdd8f8a8-0d53-4f", "
2      19843875-6077-49" );
3
4  insert into BoardMember values ( "bddd7100-cd10-41", "372a9ad5
5      -4952-44" );

```

Listing 10: BoardMember Insert Statements



```

1 insert into List values ("64251244-40f5-45", "Admin Website", 2, "
  bdd7100-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

	Title	NumComments
1	Character Design	5

```
1 SELECT TOP 1 Title, Count(*) as NumComments FROM Task, Comment
   WHERE Task.Id=Comment.TaskId GROUP BY Title ORDER BY
   NumComments DESC;
```

Listing 15: Select Most Active Task (Comments)

	Username	NumComments
1	Amy Wood	5

```
1 SELECT TOP 1 Username, Count(*) as NumComments FROM ApplicationUser
   , Comment WHERE ApplicationUser.Id=Comment.OwnerId GROUP BY
   Username ORDER BY NumComments desc;
```

Listing 16: Select Most Active User (Comments)

	Username	NumComments
1	Carlos Cooper	1

```
1 SELECT TOP 1 Username, COUNT(*) as NumComments FROM Comment,
   ApplicationUser WHERE ApplicationUser.Id = Comment.OwnerId
   GROUP BY Username ORDER BY NumComments ASC;
```

Listing 17: Select Least Active User (Comments)

	MemberId	MemberId
1	06849b4a-e875-40	372a9ad5-4952-44
2	06849b4a-e875-40	88b03172-135a-4b
3	06849b4a-e875-40	ab38ad56-b3ab-42
4	19843875-6077-49	2ea98624-82a3-46
5	19843875-6077-49	5d842729-e31d-40
6	19843875-6077-49	64ed29be-71a0-46
7	19843875-6077-49	88664860-7e36-4b
8	19843875-6077-49	ab38ad56-b3ab-42
9	19843875-6077-49	d263667b-ea46-4b
10	2ea98624-82a3-46	5d842729-e31d-40
11	2ea98624-82a3-46	64ed29be-71a0-46
12	2ea98624-82a3-46	88664860-7e36-4b
13	2ea98624-82a3-46	ab38ad56-b3ab-42
14	2ea98624-82a3-46	d263667b-ea46-4b
15	372a9ad5-4952-44	88b03172-135a-4b
16	372a9ad5-4952-44	ab38ad56-b3ab-42
17	5d842729-e31d-40	64ed29be-71a0-46
18	5d842729-e31d-40	88664860-7e36-4b
19	5d842729-e31d-40	d263667b-ea46-4b
20	64ed29be-71a0-46	88664860-7e36-4b
21	64ed29be-71a0-46	ab38ad56-b3ab-42
22	64ed29be-71a0-46	d263667b-ea46-4b
23	88664860-7e36-4b	d263667b-ea46-4b
24	88b03172-135a-4b	ab38ad56-b3ab-42

```
1 SELECT DISTINCT A.MemberId, B.MemberId FROM BoardMember AS A INNER
JOIN BoardMember as B ON A.BoardId = B.BoardId AND B.MemberId >
A.MemberId;
```

Listing 18: Find Users Who Share At Least One Board

	Username	NumBoards
1	Amy Wood	2

```
1 SELECT Username, Count(*) as NumBoards FROM BoardMember INNER JOIN
ApplicationUser on BoardMember.MemberId = ApplicationUser.Id
WHERE Username = 'Amy Wood' GROUP BY Username;
```

Listing 19: Select Board Membership for User "Amy Wood"

	MessageText	TimeCreated
1	That's literally Luigi.	2018-01-20 09:04:00.0000000 +10:00
2	Luigi?	2018-01-20 09:02:00.0000000 +10:00

```
1 SELECT TOP 10 MessageText, TimeCreated FROM Comment INNER JOIN
ApplicationUser on ApplicationUser.Id = Comment.OwnerId WHERE
Username = 'Phillip Ramirez' ORDER BY TimeCreated DESC;
```

Listing 20: Select 10 Most Recent Comments for User "Phillip Ramirez"

	MessageText	TimeCreated	Creator
1	We can discuss this in 3 we...	2018-01-22 10:40:00.0000000 +10:00	Carlos Cooper
2	If you can fly, it totally ...	2018-01-22 10:23:00.0000000 +10:00	Amy Wood
3	I mean - Isn't the idea tha...	2018-01-22 10:22:00.0000000 +10:00	Amy Wood
4	Doesn't that defeat the pur...	2018-01-22 10:21:00.0000000 +10:00	Amy Wood
5	I feel like there's some pr...	2018-01-22 10:20:00.0000000 +10:00	Amy Wood
6	Come see me, I'll show you ...	2018-01-20 09:05:00.0000000 +10:00	Roy Bailey
7	That's literally Luigi.	2018-01-20 09:04:00.0000000 +10:00	Phillip Ramirez
8	No, he'll be just like mari...	2018-01-20 09:03:00.0000000 +10:00	Roy Bailey
9	Luigi?	2018-01-20 09:02:00.0000000 +10:00	Phillip Ramirez
10	So I'm thinking our game ha...	2018-01-20 08:54:00.0000000 +10:00	Roy Bailey

```
1 SELECT TOP 10 MessageText, TimeCreated, Username as Creator FROM
Comment, ApplicationUser WHERE ApplicationUser.Id = Comment.
OwnerId ORDER BY TimeCreated DESC;
```

Listing 21: Select 10 Most Recent Comments

	Board	NumTasks
1	AI Stock Trader	12

```
1 SELECT TOP 1 Board.Title as Board, COUNT(*) as NumTasks FROM Task
INNER JOIN List ON Task.ListId = List.Id INNER JOIN Board on
List.BoardId = Board.Id GROUP BY Board.Title ORDER BY NumTasks
DESC;
```

Listing 22: Select Most Active Board (Tasks)

---

	Board	NumComments
1	Untitled Platformer Game	10

```
1 SELECT TOP 1 Board.Title as Board, COUNT(*) as NumComments FROM
    Comment INNER JOIN Task on Comment.TaskId = Task.Id INNER JOIN
    List ON Task.ListId = List.Id INNER JOIN Board on List.BoardId
    = Board.Id GROUP BY Board.Title ORDER BY NumComments DESC;
```

Listing 23: Select Most Active Board (Comments)

---

	Username	AssignedTasks
1	Amy Wood	3

```
1 SELECT TOP 1 Username, COUNT(*) AS AssignedTasks FROM
    ApplicationUser INNER JOIN TaskAssignee on ApplicationUser.Id =
    TaskAssignee.AssigneeId GROUP BY Username ORDER BY
    AssignedTasks DESC;
```

Listing 24: Select Most Active User (Tasks)

---

	Username	BoardsCreated
1	Walter Rogers	2

```
1 SELECT TOP 1 Username, COUNT(*) AS BoardsCreated FROM
    ApplicationUser INNER JOIN Board on ApplicationUser.Id = Board.
    OwnerId GROUP BY Username ORDER BY BoardsCreated DESC;
```

Listing 25: Select Most Active User (Boards)

---

	Title
1	Choose a platform.
2	Evaluation Metrics
3	Logging
4	Continuous Gathering
5	Interaction
6	Launch
7	Clean Data
8	Moderators
9	Level Design
10	Model Selection
11	Continuous Deployment
12	Training
13	Develop
14	Web App
15	Database
16	Player Mechanics
17	Deployment
18	Landscape Design
19	Research
20	Packaging
21	Character Design
22	Control
23	Data Formatting
24	Implementation
25	Story Design
26	Database Design
27	Testing
28	Enemies
29	Marketing Strategy
30	GUI

```
1 SELECT Id FROM Task WHERE DueDate < SYSDATETIMEOFFSET();
```

Listing 26: Select All Overdue Tasks

### 2.3.2 Other Statements

### 2.3.3 Update statements

## 3 Index

### Listings

1	ApplicationUser Table Creation Statement . . . . .	6
2	Board Table Creation Statement . . . . .	6
3	BoardMember Table Creation Statement . . . . .	6
4	List Table Creation Statement . . . . .	6
5	Task Table Creation Statement . . . . .	6
6	TaskAssignee Table Creation Statement . . . . .	7
7	Comment Table Creation Statement . . . . .	7
8	ApplicationUser Insert Statements . . . . .	7
9	Board Insert Statements . . . . .	7
10	BoardMember Insert Statements . . . . .	7
11	List Insert Statements . . . . .	8
12	Task Insert Statements . . . . .	8
13	TaskAssignee Insert Statements . . . . .	8

14	Comment Insert Statements . . . . .	8
15	Select Most Active Task (Comments) . . . . .	9
16	Select Most Active User (Comments) . . . . .	9
17	Select Least Active User (Comments) . . . . .	9
18	Find Users Who Share At Least One Board . . . . .	10
19	Select Board Membership for User "Amy Wood" . . . . .	10
20	Select 10 Most Recent Comments for User "Phillip Ramirez" . . . . .	10
21	Select 10 Most Recent Comments . . . . .	10
22	Select Most Active Board (Tasks) . . . . .	10
23	Select Most Active Board (Comments) . . . . .	11
24	Select Most Active User (Tasks) . . . . .	11
25	Select Most Active User (Boards) . . . . .	11
26	Select All Overdue Tasks . . . . .	12