

"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

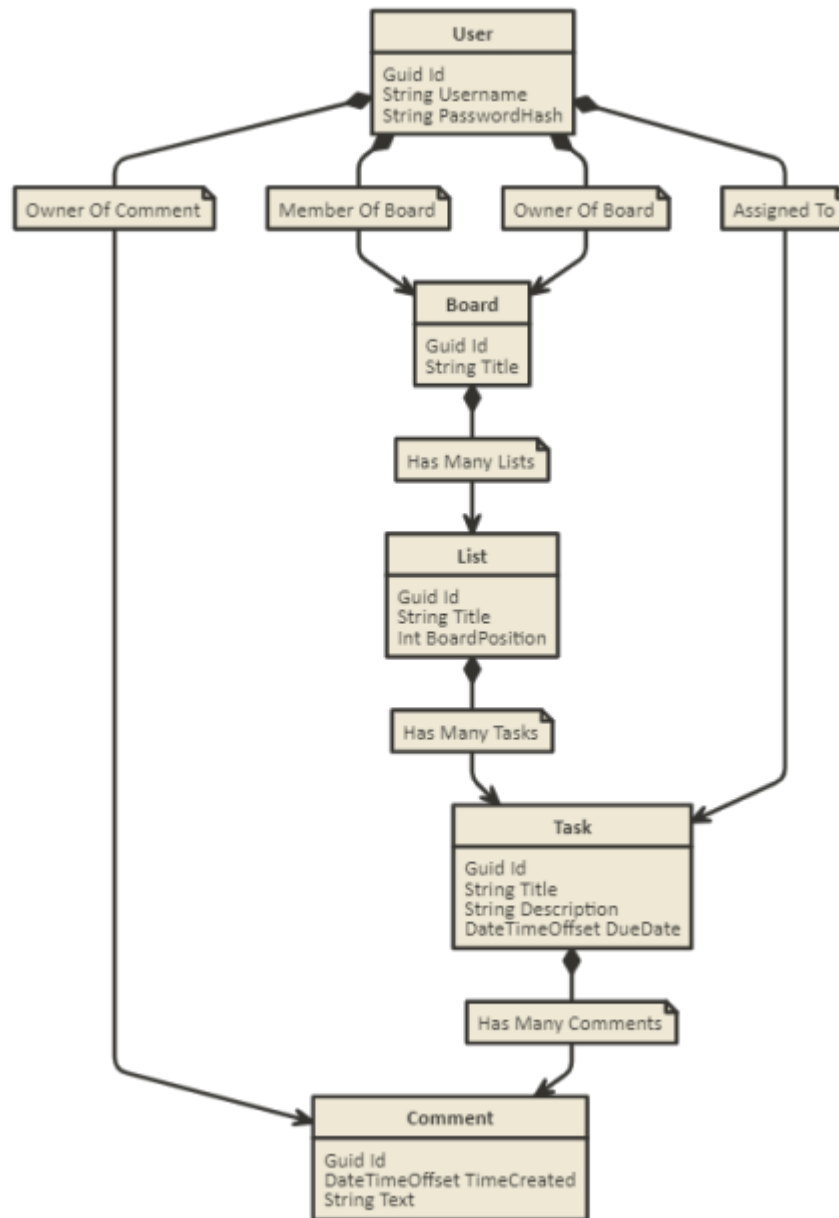
1	Database Design	2
1.1	Introduction	2
1.2	ER Diagram	3
1.3	Identified Constraints	4
1.4	Assumptions About the Domain	4
1.5	Database Design Process	4
1.6	Ollert's tables	4
2	Database Implementation	6
2.1	Create Table Statements	6
2.2	Insert Statements	7
2.3	Data Tables	9
2.4	Data manipulation statements	15
3	Index	19

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 Tables

Due to the length of some of our fields the results may be split into multiple tables to fit on the page. Tables in the same horizontal rule pair are part of the same table.

ApplicationUser Table

Id	Username
06849b4a-e875-40	Rebecca Moore
19843875-6077-49	Walter Rogers
2ea98624-82a3-46	Marie Ross
372a9ad5-4952-44	Jean Bryant
5d842729-e31d-40	Carlos Cooper
64ed29be-71a0-46	Amy Wood
88664860-7e36-4b	Phillip Ramirez
88b03172-135a-4b	Samuel Barnes
ab38ad56-b3ab-42	Frank Bell
d263667b-ea46-4b	Roy Bailey
ecff8345-aa8b-40	Jesse Richardson

Passwordhash

4DCBF8F1C8BEF27458F1BD8DED724AFF3182C712057CE3BA11DFCF170244AA92168E5F6A717237FB2232A8AFE2DAAE3F8D582C5D4CC0EAA268F05F420F1EC421661F9EC03BD2D1D8D9D9BC08B2B71B414C6DDBD31B8ECE132DC9C112070B026EDAB12D7BB613EAC0304D9917738729FB37B60EBB1FB59FC9493ED64733CCE3BAF892F6A6B56B2B5079940B59E423D5E91CE28ED45A925CA9E5B29C2757A00E2645C12F6E3334CDFCDD2DC245339D8221C430EE7EA8F93291529B27536039878765CB325076C4EA9F34927D5C45D013AC89A6985C5CD84DF07EB3BEB0322362715138B68F2AA87B19781E84B1E6FA0EA1CDC963620FE30343385C118B17AFA7DA9B0746CF155968BA110AF6B8791940DDD3AC7253BBFB835482C33150C34CA452F269CC6E9580A6F3D7AED20805D3C21EBF79C18BEDF5D930210966651FB40370ABB1C197D22CC1017A9471C46BA7C9021A98F4D7F77E457063DCB75ED6814837
--

Board Table

Id	Title	OwnerId
62376bd0-6ecc-4f	Untitled Platformer Game	372a9ad5-4952-44
bbdd7100-cd10-41	Sports Forum Mobile App	19843875-6077-49
fdd8f8a8-0d53-4f	AI Stock Trader	19843875-6077-49

Board Member Table

BoardId	MemberId
fdd8f8a8-0d53-4f	19843875-6077-49
bbdd7100-cd10-41	372a9ad5-4952-44
62376bd0-6ecc-4f	d263667b-ea46-4b
fdd8f8a8-0d53-4f	64ed29be-71a0-46
bbdd7100-cd10-41	88b03172-135a-4b
62376bd0-6ecc-4f	88664860-7e36-4b
fdd8f8a8-0d53-4f	2ea98624-82a3-46
bbdd7100-cd10-41	06849b4a-e875-40
62376bd0-6ecc-4f	5d842729-e31d-40
fdd8f8a8-0d53-4f	ab38ad56-b3ab-42
bbdd7100-cd10-41	ab38ad56-b3ab-42
62376bd0-6ecc-4f	2ea98624-82a3-46
62376bd0-6ecc-4f	64ed29be-71a0-46
62376bd0-6ecc-4f	19843875-6077-49
fdd8f8a8-0d53-4f	19843875-6077-49
bbdd7100-cd10-41	372a9ad5-4952-44
62376bd0-6ecc-4f	d263667b-ea46-4b
fdd8f8a8-0d53-4f	64ed29be-71a0-46
bbdd7100-cd10-41	88b03172-135a-4b
62376bd0-6ecc-4f	88664860-7e36-4b
fdd8f8a8-0d53-4f	2ea98624-82a3-46
bbdd7100-cd10-41	06849b4a-e875-40
62376bd0-6ecc-4f	5d842729-e31d-40
fdd8f8a8-0d53-4f	ab38ad56-b3ab-42
bbdd7100-cd10-41	ab38ad56-b3ab-42
62376bd0-6ecc-4f	2ea98624-82a3-46
62376bd0-6ecc-4f	64ed29be-71a0-46
62376bd0-6ecc-4f	19843875-6077-49

List Table

Id	Title	BoardPosition	BoardId
0e72d679-da23-41	Database	0	bbdd7100-cd10-41
3073d9ee-fbef-48	ML Model Creation	1	fdd8f8a8-0d53-4f
42f550ff-724b-49	Deployment	3	fdd8f8a8-0d53-4f
64251244-40f5-45	Admin Website	2	bbdd7100-cd10-41
89fd7d40-e284-44	Game Development	2	62376bd0-6ecc-4f
8dda7e83-294f-48	Logviewer	2	fdd8f8a8-0d53-4f
a583557a-3d83-4a	Art Design	0	62376bd0-6ecc-4f
b9ec9dee-e742-4d	App Development	1	bbdd7100-cd10-41
d0ce1983-dcdf-49	Distribution	1	62376bd0-6ecc-4f
ed04c7f0-20e3-48	Data Gathering	0	fdd8f8a8-0d53-4f

Task Table

Id	Title
00de54ec-ff13-4e	Choose a platform.
03aa56ed-84c3-4d	Evaluation Metrics
04cac4f2-1ea7-48	Logging
0fc5c4d2-019c-49	Continous Gathering
15c31e03-c080-43	Interaction
177bb237-60ab-47	Launch
18e4c14b-56d6-43	Clean Data
199ec36c-bdea-40	Moderators
20de062c-f25e-46	Level Design
323c8fb9-e7c6-44	Model Selection
46202c62-3c33-46	Continous Deployment
4eb0744b-f9d2-49	Training
56378d8b-0c82-4a	Develop
70cad687-5f05-46	Web App
798be7dd-61c2-4d	Database
8ada3afb-5fd3-43	Player Mechanics
8cdd71f6-ed8b-45	Deployment
8cee3017-a75f-4a	Landscape Design
95a5ad82-6316-42	Research
99049f5b-3d88-4c	Packaging
a9f5ae59-6193-40	Character Design
ac1a2b65-d1fc-47	Control
b81246d7-2bf0-49	Data Formatting
bc63d228-b7a9-49	Implementation
d46993c0-ce64-46	Story Design
e01479d6-6019-4a	Database Design
e06959ed-e216-4f	Testing
e1381441-265b-46	Enemies
f7b17de0-c269-43	Marketing Strategy
f8acf56a-4860-4c	GUI

Descriptor
<p>Develop metrics to help select the model.</p> <p>Develop an application that uses the ML Model to log good stock trades.</p> <p>Come up with a way to continuously gather data.</p> <p>Create a Web API for interacting with the database.</p> <p>Establish a distribution pipeline and ensure it can handle demand.</p> <p>Develop a methodology for discarding the extraneous or bad data.</p> <p>Create a business process for choosing moderators.</p> <p>Design a variety of levels for the player to play through.</p> <p>Select the ML Model that will work best.</p>

Descriptor
Develop a process for deploying updates to live applications.
Experiment and find the best way to train the model.
Develop the application according to requirements gathered.
Create a web application w/ secure login for performing administrative actions.
Develop a database to store logs.
Develop interesting and challenging mechanics for the player to interact with.
Get the app approved for distribution in the app stores.
Create interesting environments/backdrops for the game to take place in.
Research market demographics to find an audience for the game.
Find out how to package all parts of the app together.
Create the designs for the characters.
Implement interactions with the database for basic administration controls.
Format the data for use with our chosen model.
Choose a SQL Implementation and write create statements for the tables.
Write a fun story for the game.
Design a robust database schema for storing all the data in our app.
Develop a robust testing suite.
Develop Enemy AIs that interact with the player and the environment in a variety of ways.
Design a marketing strategy to target key demographics.
Create a GUI for viewing the logs.

DueDate	ListId
1/20/2018 9:00:00 AM +10:00	b9ec9dee-e742-4d
2/28/2018 9:00:00 AM +10:00	3073d9ee-fbef-48
3/20/2018 9:00:00 AM +10:00	8dda7e83-294f-48
1/25/2018 9:00:00 AM +10:00	ed04c7f0-20e3-48
3/20/2018 9:00:00 AM +10:00	0e72d679-da23-41
6/20/2018 9:00:00 AM +10:00	d0ce1983-dcdf-49
1/20/2018 9:00:00 AM +10:00	ed04c7f0-20e3-48
3/20/2018 9:00:00 AM +10:00	64251244-40f5-45
5/20/2018 9:00:00 AM +10:00	89fd7d40-e284-44
2/20/2018 9:00:00 AM +10:00	3073d9ee-fbef-48
4/20/2018 9:00:00 AM +10:00	42f550ff-724b-49
2/28/2018 9:00:00 AM +10:00	3073d9ee-fbef-48
3/20/2018 9:00:00 AM +10:00	b9ec9dee-e742-4d
3/20/2018 9:00:00 AM +10:00	64251244-40f5-45
1/20/2018 9:00:00 AM +10:00	8dda7e83-294f-48
4/20/2018 9:00:00 AM +10:00	89fd7d40-e284-44
4/20/2018 9:00:00 AM +10:00	b9ec9dee-e742-4d
6/20/2018 9:00:00 AM +10:00	a583557a-3d83-4a
4/20/2018 9:00:00 AM +10:00	d0ce1983-dcdf-49
4/20/2018 9:00:00 AM +10:00	42f550ff-724b-49
3/20/2018 9:00:00 AM +10:00	a583557a-3d83-4a
3/20/2018 9:00:00 AM +10:00	64251244-40f5-45

DueDate	ListId
1/25/2018 9:00:00 AM +10:00	ed04c7f0-20e3-48
2/20/2018 9:00:00 AM +10:00	0e72d679-da23-41
6/20/2018 9:00:00 AM +10:00	a583557a-3d83-4a
1/20/2018 9:00:00 AM +10:00	0e72d679-da23-41
4/20/2018 9:00:00 AM +10:00	42f550ff-724b-49
5/20/2018 9:00:00 AM +10:00	89fd7d40-e284-44
6/20/2018 9:00:00 AM +10:00	d0ce1983-dcdf-49
3/20/2018 9:00:00 AM +10:00	8dda7e83-294f-48

TaskAssignee Table

TaskId	AssigneeId
18e4c14b-56d6-43	19843875-6077-49
323c8fb9-e7c6-44	64ed29be-71a0-46
798be7dd-61c2-4d	2ea98624-82a3-46
99049f5b-3d88-4c	ab38ad56-b3ab-42
e01479d6-6019-4a	372a9ad5-4952-44
e01479d6-6019-4a	88b03172-135a-4b
199ec36c-bdea-40	06849b4a-e875-40
bc63d228-b7a9-49	ab38ad56-b3ab-42
56378d8b-0c82-4a	372a9ad5-4952-44
ac1a2b65-d1fc-47	88b03172-135a-4b
a9f5ae59-6193-40	d263667b-ea46-4b
d46993c0-ce64-46	88664860-7e36-4b
8cee3017-a75f-4a	5d842729-e31d-40
95a5ad82-6316-42	2ea98624-82a3-46
f7b17de0-c269-43	64ed29be-71a0-46
177bb237-60ab-47	64ed29be-71a0-46

Comment Table

Id	TimeCreated	
2151adf7-6de1-40	1/10/2018 9:25:00 AM +10:00	
49570bdb-0702-45	1/10/2018 9:55:00 AM +10:00	
4e8a2544-f829-4e	1/20/2018 9:04:00 AM +10:00	
59646b1b-a084-49	1/20/2018 8:54:00 AM +10:00	
5f71a133-2371-40	1/20/2018 9:03:00 AM +10:00	
990d6b06-7759-42	1/20/2018 9:05:00 AM +10:00	
99487abe-6ccd-45	1/20/2018 9:02:00 AM +10:00	
9aeb90fd-8f48-4b	1/10/2018 9:54:00 AM +10:00	
bccb9a9a-0566-49	1/15/2018 9:00:00 AM +10:00	
c088cfa0-15e2-4f	1/10/2018 8:54:00 AM +10:00	
e6a99415-8607-46	1/10/2018 9:00:00 AM +10:00	

MessageText
<p>Cant start packaging stuff up until we know what were gonna have to package. Ive got some blockers on this task. I mean - Isnt the idea that you JUMP from platform to platform? thx m8 Thats literally Luigi. So Im thinking our game has a mario-like character - but green. I feel like theres some problems with having the main character fly. Luigi? Definitely. Ill help you clean it up. Can someone explain this Task to me? Im probably going to need some help with this. Doesnt that defeat the purpose of a platformer? Never mind.</p>

TaskId	OwnerId
99049f5b-3d88-4c	ab38ad56-b3ab-42
e01479d6-6019-4a	88b03172-135a-4b
199ec36c-bdea-40	06849b4a-e875-40
8ada3afb-5fd3-43	64ed29be-71a0-46
8ada3afb-5fd3-43	64ed29be-71a0-46
18e4c14b-56d6-43	19843875-6077-49
a9f5ae59-6193-40	88664860-7e36-4b
a9f5ae59-6193-40	d263667b-ea46-4b
a9f5ae59-6193-40	d263667b-ea46-4b
8ada3afb-5fd3-43	64ed29be-71a0-46
8ada3afb-5fd3-43	5d842729-e31d-40
a9f5ae59-6193-40	d263667b-ea46-4b
a9f5ae59-6193-40	88664860-7e36-4b
18e4c14b-56d6-43	64ed29be-71a0-46
8cdd71f6-ed8b-45	ab38ad56-b3ab-42
04cac4f2-1ea7-48	2ea98624-82a3-46
e01479d6-6019-4a	372a9ad5-4952-44
8ada3afb-5fd3-43	64ed29be-71a0-46
8cdd71f6-ed8b-45	ab38ad56-b3ab-42
18e4c14b-56d6-43	19843875-6077-49

2.4 Data manipulation statements

2.4.1 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

2.4.2 Update statements

```
1 UPDATE Comment SET MessageText = 'Thanks mate.', TimeCreated =
  SYSDATETIMEOFFSET() WHERE Id = '49570bdb-0702-45';
```

Listing 27: Edit Comment Text

```
1 UPDATE Board SET OwnerId = '5d842729-e31d-40' WHERE Title = 'AI
  Stock Trader';
```

Listing 28: Change Board Owner

```
1 UPDATE Task SET Descriptor = 'Develop a robust and effective
  testing suite.' WHERE Title = 'Testing';
```

Listing 29: Change Task Description

```
1 UPDATE List SET Title = 'Art & Design' WHERE Title = 'Art Design';
```

Listing 30: Rename a List

2.4.3 Other Statements

```
1 DELETE FROM Comment WHERE TimeCreated < '2018-01-12';
```

Listing 31: Delete All Comments Prior to A Given Date

```

1 DELETE FROM ApplicationUser WHERE Id NOT IN (SELECT OwnerId FROM
    Board UNION SELECT MemberId FROM BoardMember UNION SELECT
    OwnerId FROM Comment UNION SELECT AssigneeId FROM
    TaskAssignee);

```

Listing 32: Delete Inactive Users

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)	15
16	Select Most Active User (Comments)	15
17	Select Least Active User (Comments)	15
18	Find Users Who Share At Least One Board	16
19	Select Board Membership for User "Amy Wood"	16
20	Select 10 Most Recent Comments for User "Phillip Ramirez" . .	16
21	Select 10 Most Recent Comments	16
22	Select Most Active Board (Tasks)	16
23	Select Most Active Board (Comments)	17
24	Select Most Active User (Tasks)	17
25	Select Most Active User (Boards)	17
26	Select All Overdue Tasks	18
27	Edit Comment Text	18
28	Change Board Owner	18
29	Change Task Description	18
30	Rename a List	18
31	Delete All Comments Prior to A Given Date	18
32	Delete Inactive Users	19