

Penpal with me

Database Management
System
19IS4PCDBM
Nalina V

H Nidhi
1BM19IS056
Meghana Ratharaj
1BM19IS087

Introduction

- A penpal is someone with whom you communicate via letters. They are usually strangers whose relationship is based primarily, or even solely, on their exchange of letters / emails.
- “Penpal with me” is an online platform which enables people to find penpals based on a set of similar interests and at least one common language.

A decorative map of Norway is visible in the background, with the landmass in light green and surrounding waters in light blue. The map is partially obscured by the text and other elements.

Problem definition

Lack of online tools to connect with people of similar interests and common language during the pandemic for the purpose of long form communication

Functionality

- Sign-up and Login to existing account
- Update password
- Update interests
- Update languages
- Add new penpal
- Send & receive messages
- Last sent message interval
- Delete penpal
- Multiple penpals
- Display penpals

Requirements

Software Requirements

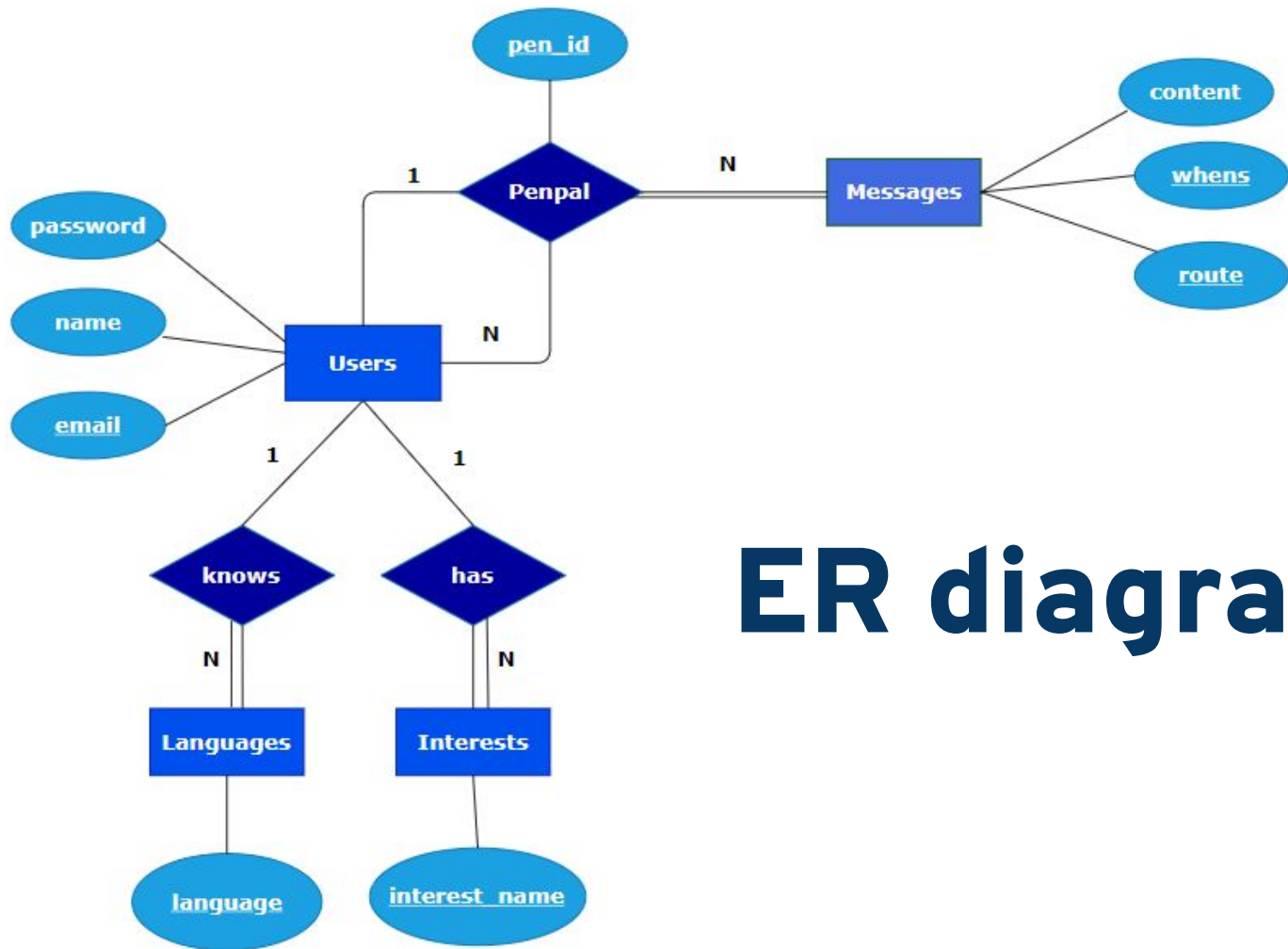
- Operating System: Windows/Linux
- Backend: XAMPP Server
- Front end tool: Any browser
- Database: MariaDB with XAMPP

Hardware Requirements

- Processor: Pentium IV
- Hard Disk: 40 GB
- Ram: 512 MB

Tools used

- Bootstrap 5.0
- Github
- VS Code
- XAMPP server
- ER diagram maker
draw.io
- Google slides



ER diagram

ER diagram to Relational Schema

Users(name, email, password)

- User -> speaks -> language (1:N relationship)

Hence, we can combine the relationship table & language

Languages(email, language)

- User -> has -> interests (1:N relationship)

Hence, we can combine the relationship table & interests

Interests(email, interest)

- User -> penpal -> user (recursive relationship)

This recursive relationship cannot be reduced further

Penpals(pen_id, user1, user2)

- User -> penpal -> messages (M:N relationship)

This relationship is many-to-many, hence cannot be reduced

Messages(message_id, pen_id, sender, content, when)

Relational Schema

Users(name, email, password)

Interests(email, interest)

Languages(email, language)

Penpals(pen_id, user1, user2)

Messages(message_id, pen_id, sender, content, when)

- **Users** table wasn't in 3NF
This was solved by making email as the PK
 - Allow names to be repetitive
Hence, name cannot be a determining factor
- **Messages** wasn't in 3NF because of a transitive FD
 - Causes column level redundancy, update anomaly
 - This was solved by decomposing the table

Decomposing to 3NF

Messages(message_id, pen_id, sender, content, when) isn't in 3NF because

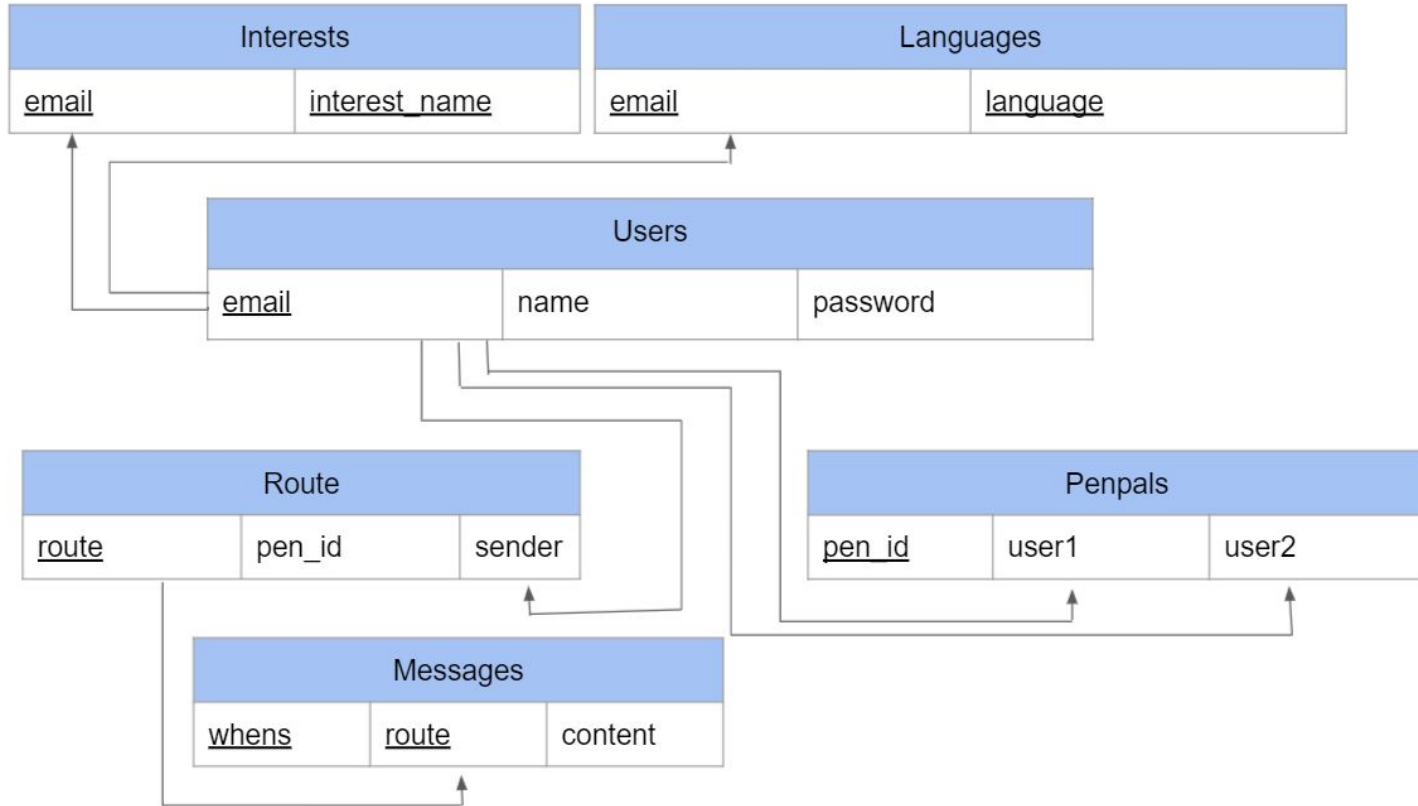
- (pen_id, sender, when) becomes a CK
- (sender, when) also becomes a CK
- When we choose message_id as PK,
 - (sender, when) become NPA
 - Hence, NPA -> NPA causes a transitive FD
- Causes column level redundancy, because for message sent, the (pen_id, sender) gets repeated
 - Update anomaly, updating pen_id of a pair will be bulk updation, can cause inconsistency

Messages(route_id, content, when)

Route(route_id, pen_id, sender)

This decomposition satisfies 3NF + avoids column level redundancy

Relational Schema



Highest normal form

1) Users (email, password, name);

Candidate keys : email

2) Interests(email, interest);

Candidate keys : (email, interest)

3) Languages (email, language);

Candidate keys : (email, language)

4) Penpals (pen_id, user1, user2);

Candidate keys : pen_id

5) Messages (content, whens, route);

Candidate keys : (whens, route)

6) Route (route, pen_id, sender);

Candidate keys : route

All tables are BCNF form!

Tables

```
mysql> DESC Users;
```

Field	Type	Null	Key	Default	Extra
email	varchar(50)	NO	PRI	NULL	
name	varchar(50)	YES		NULL	
password	varchar(50)	YES		NULL	

```
3 rows in set (0.77 sec)
```

```
mysql> DESC Languages;
```

Field	Type	Null	Key	Default	Extra
email	varchar(50)	NO	PRI	NULL	
language	varchar(50)	NO	PRI	English	

```
2 rows in set (0.00 sec)
```

```
mysql> DESC Interests;
```

Field	Type	Null	Key	Default	Extra
email	varchar(50)	NO	PRI	NULL	
interest_name	varchar(50)	NO	PRI	Random	

```
2 rows in set (0.00 sec)
```

Tables

```
mysql> DESC Penpals;
```

Field	Type	Null	Key	Default	Extra
pen_id	int	NO	PRI	NULL	auto_increment
user1	varchar(50)	NO	MUL	NULL	
user2	varchar(50)	NO	MUL	NULL	

```
3 rows in set (0.00 sec)
```

```
mysql> DESC Route;
```

Field	Type	Null	Key	Default	Extra
route	int	NO	PRI	NULL	auto_increment
pen_id	int	NO	MUL	NULL	
sender	varchar(50)	YES	MUL	NULL	

```
3 rows in set (0.01 sec)
```

```
mysql> DESC Messages;
```

Field	Type	Null	Key	Default	Extra
whens	datetime	NO	PRI	CURRENT_TIMESTAMP	DEFAULT_GENERATED
route	int	NO	PRI	NULL	
content	text	YES		NULL	

```
3 rows in set (0.00 sec)
```

Implementation

```
CREATE TABLE Users (  
  email VARCHAR(50) NOT NULL,  
  name VARCHAR(50),  
  password VARCHAR(50),  
  PRIMARY KEY (email)  
);
```

```
CREATE TABLE Interests(  
  email VARCHAR(50) NOT NULL,  
  interest_name VARCHAR(50) DEFAULT 'Random',  
  PRIMARY KEY (email, interest_name),  
  FOREIGN KEY (email) REFERENCES Users (email)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE  
);
```

```
CREATE TABLE Languages (  
  email VARCHAR(50) NOT NULL,  
  language VARCHAR(50) DEFAULT 'English',  
  PRIMARY KEY (email, language),  
  FOREIGN KEY (email) REFERENCES Users (email)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE  
);
```


Implementation

```
CREATE TABLE Penpals (  
  pen_id INT NOT NULL AUTO_INCREMENT,  
  user1 VARCHAR(50) NOT NULL,  
  user2 VARCHAR(50) NOT NULL,  
  PRIMARY KEY (pen_id),  
  FOREIGN KEY (user1) REFERENCES ...,  
  FOREIGN KEY (user2) REFERENCES ...  
);
```

```
CREATE TABLE Route (  
  route INT NOT NULL AUTO_INCREMENT,  
  pen_id INT NOT NULL,  
  sender VARCHAR(50),  
  PRIMARY KEY (route),  
  FOREIGN KEY (pen_id) REFERENCES ...,  
  FOREIGN KEY (sender) REFERENCES ...  
);
```

```
CREATE TABLE Messages (  
  whens DATETIME DEFAULT CURRENT_TIMESTAMP,  
  route INT NOT NULL,  
  content TEXT(1000),  
  PRIMARY KEY (route, whens),  
  FOREIGN KEY (route) REFERENCES Route (route)  
);
```


Implementation

INSERT INTO Users VALUES(?, ?, ?);

INSERT INTO Interests(email) VALUES (?);

INSERT INTO Languages(email) VALUES (?);

INSERT INTO Penpals(user1,user2) VALUES (?, ?);

INSERT INTO Route(pen_id,sender) VALUES (?, ?);

INSERT INTO Messages(route,content) VALUES (?, ?);

INSERT INTO Messages(route,content) VALUES (?, ?);

INSERT INTO Interests VALUES (?, ?);

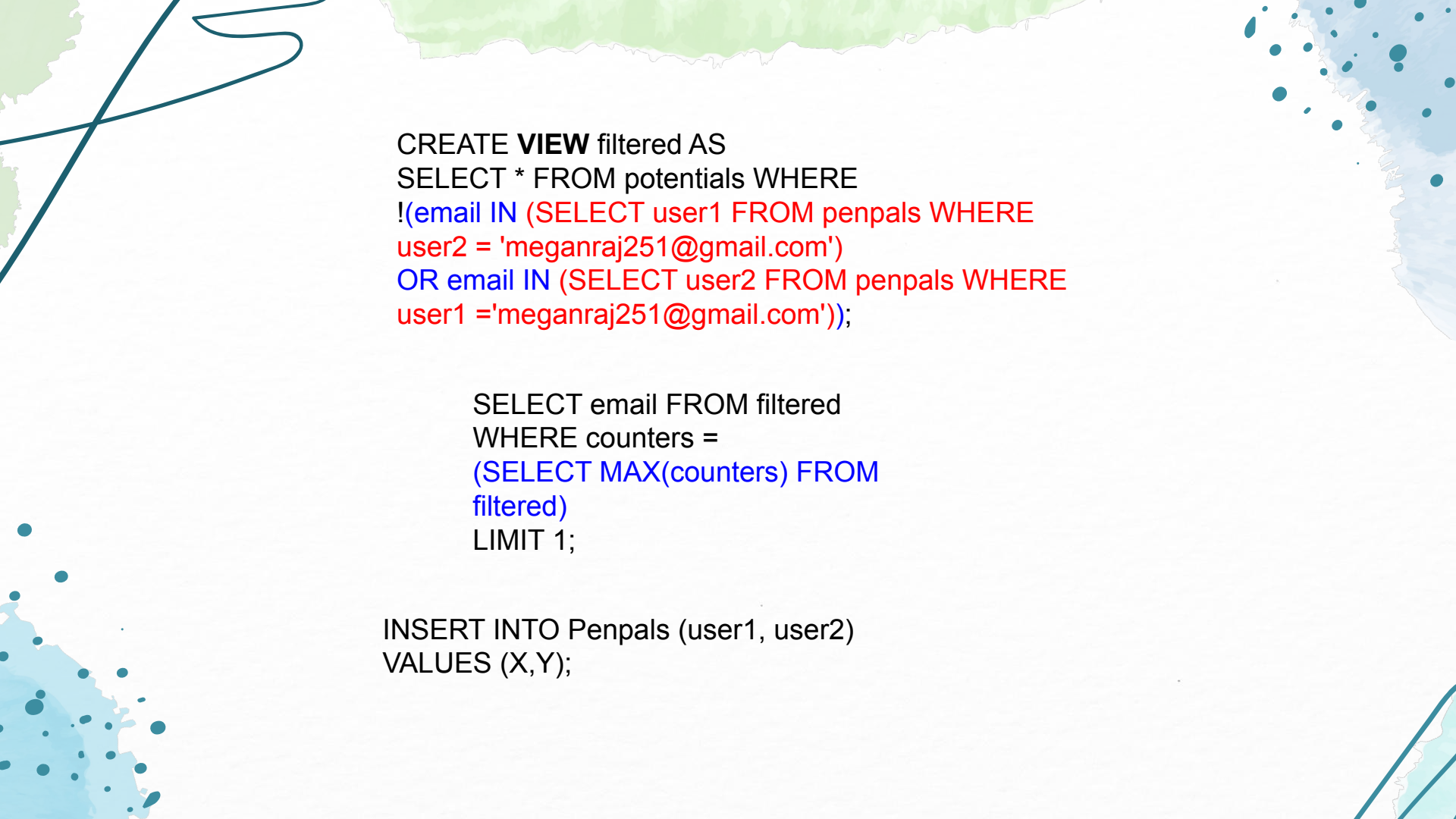
INSERT INTO Languages VALUES (?, ?);

Implementation

Filtering a new penpal

```
CREATE VIEW my_pals AS
SELECT y.email, MAX(y.num)-1 AS counters
FROM
(SELECT COUNT(email) AS num, email
FROM Interests
WHERE interest_name IN
(SELECT interest_name FROM Interests
WHERE email = 'meganraj251@gmail.com')
AND email <> 'meganraj251@gmail.com'
GROUP BY email) y
GROUP BY email
HAVING MAX(y.num) != 1
ORDER BY counters DESC;
```

```
CREATE VIEW potentials AS
SELECT DISTINCT my_pals.email, my_pals.counters
FROM
my_pals NATURAL JOIN Languages
WHERE language IN
(SELECT language FROM Languages
WHERE email = 'meganraj251@gmail.com');
```



```
CREATE VIEW filtered AS  
SELECT * FROM potentials WHERE  
!(email IN (SELECT user1 FROM penpals WHERE  
user2 = 'meganraj251@gmail.com')  
OR email IN (SELECT user2 FROM penpals WHERE  
user1 ='meganraj251@gmail.com'));
```

```
SELECT email FROM filtered  
WHERE counters =  
(SELECT MAX(counters) FROM  
filtered)  
LIMIT 1;
```

```
INSERT INTO Penpals (user1, user2)  
VALUES (X,Y);
```

Conclusion

Our application 'Penpal with me' has the ability :-

- ★ It allows the users to find another user as a penpal based on common interests
- ★ It allows the users to view and send messages to penpals.
- ★ It allows the users to delete penpals
- ★ It allows the users to update interests and language preferences

Future enhancements

Grouping penpals

Building a healthy community

Safety features

Allow users to report & block penpals

Multimedia

Image & videos sharing facilities

Reminders

Users can get reminders to text back their penpals

Notifications

Notify a user when they get a new message

A stylized map of Indonesia is shown in the background. The landmasses are colored in various shades of green, and the surrounding water is light blue. There are some small dark blue dots scattered on the map, possibly representing islands or specific locations. The text "Thank You" is prominently displayed in the center of the map.

Thank You

::)