CS5200 Project Proposal
Yaming Zhang

1. **Group name**: ZhangY with only one member Yaming Zhang
2. **Description of the project**:

   I would like to design a system for travel or trip records, which could help record your trips with your friends and families! These days traveling has become more and more normal for college students and adults. However, there is no convenient application for us to remember all the places we have gone with our beloved. So, I came up with this idea to help us remember all those precious times.

   My system is designed to have a login system, which could be used for each user to login and get their data. Users will have their own email, password, username, etc. For each user, they could search for their friends by email or username to add friends and share the information with their friends. After login, users could add their trips that they have been to. The trips consist of the departure and return date, the city and their companies. If they like, they could also add their favorite cuisines during the trip so that people could learn more about the city and restaurant recommendations.

   As I mentioned above, the main part of my application would reside on the users and the trip histories. This is also the part where I would demonstrate my ability on the CRUD operation on data as well as the procedures, functions, triggers etc.

   For extra credits, I would like to add some notification part and file upload part. For example, users could receive email notifications when they are added to the trips of their companies. For the file upload, the system may allow the users to upload the photos of the trip in their zone. I will use these parts as extra functions and extra credits.

3. **Data Storage**: For data storage, I will use MySql as my relational database.
4. **Technical Stack**:

I will use **Python** and **Django** as my backend language and RESTful framework because Django is really friendly for full stack development. I could use the template offered by Django for my frontend view display. Moreover, Django provides database ORM for MySql, which is more convenient and secure.

I would like to use **JWT** for my user authentication to ensure the security of my backend APIs.

After the development of the project, I will deploy the server on a public machine and resolve the public IP to my own domain so that people could get access to the application from all around the world.

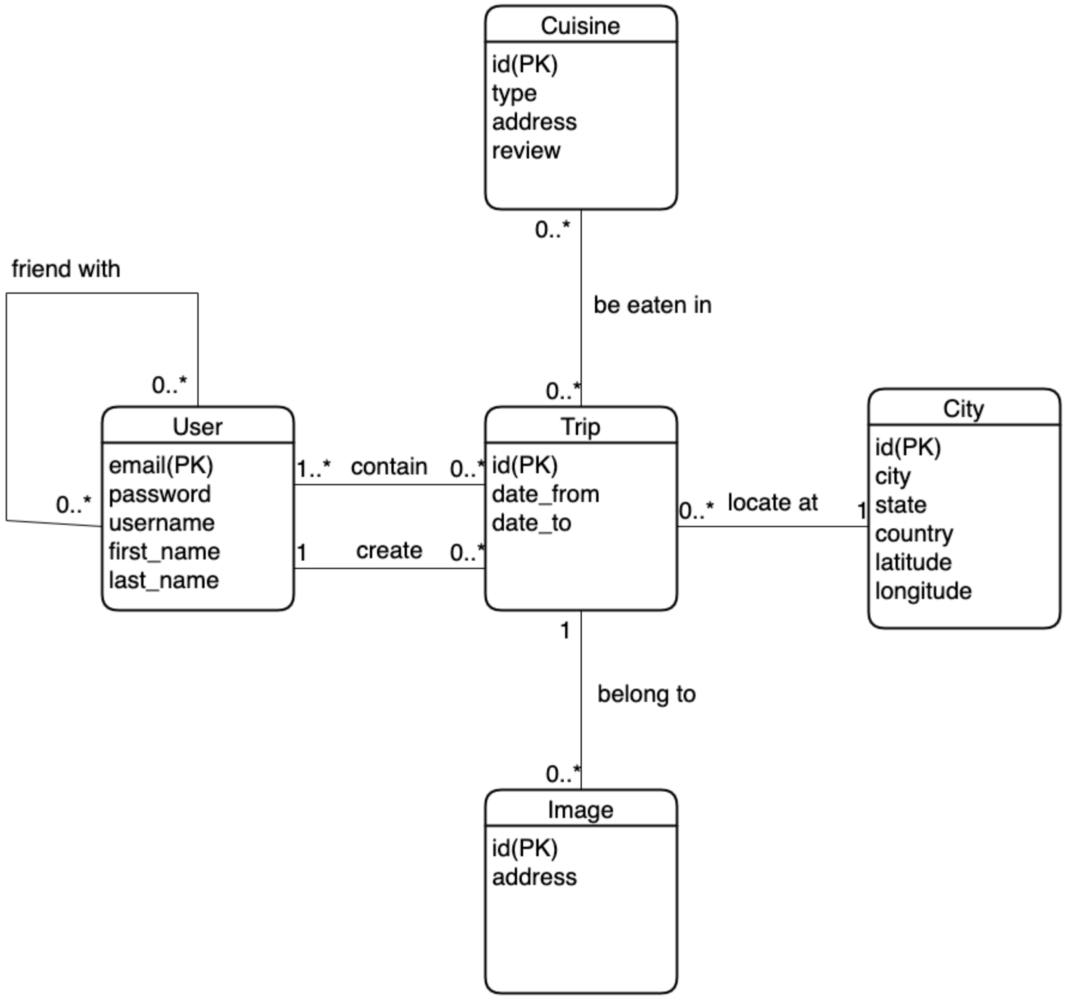
I am thinking of using some frontend libraries to show the places on the map using **GIS** features of the data. However, this part could be a little demanding for me thus I will put this part as extra effort too.

5. **Why this**?

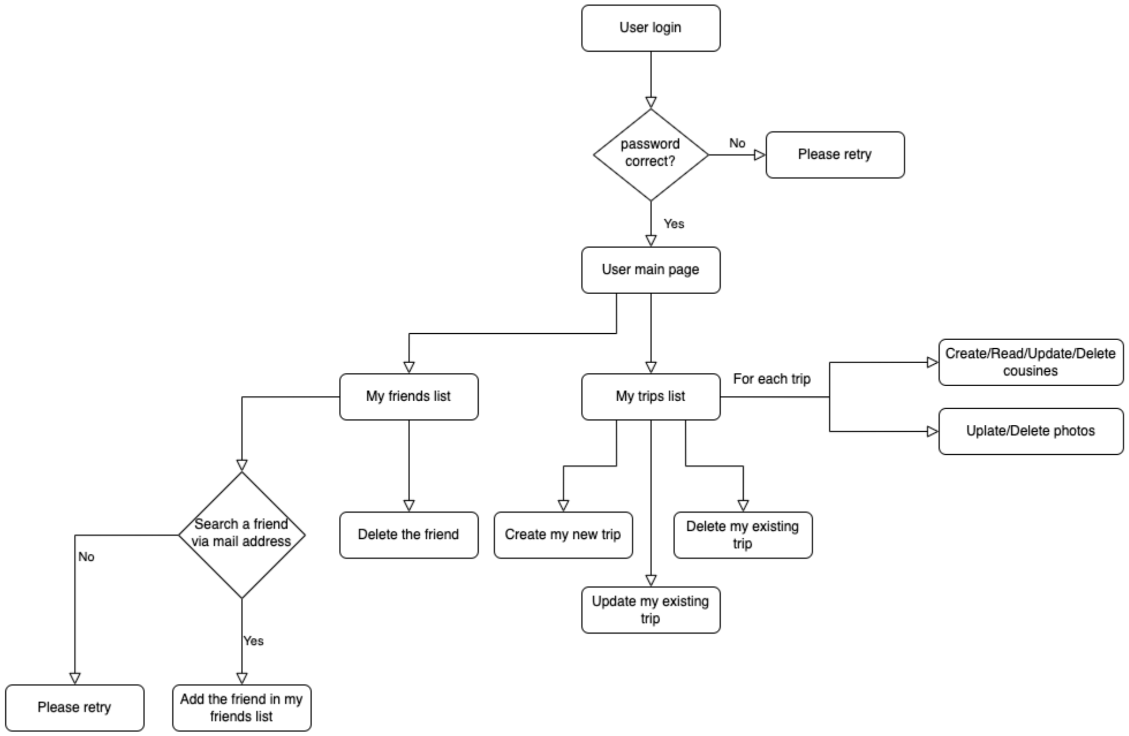
   My inspiration came for the Zenly mobile application which has been shutted down these days. Zenly is used for sharing real-time locations of your friends and families. After being shutted down, people are trying to find an alternative for the application. At that time, the idea got into my mind like I could design an application to share the trips with our friends. Although the location sharing is not real-time, we could always memorize those happy times.

   Another reason is that I have done some Geospatial data research for my computer science study. Thus, I have a huge passion for geospatial data, which could be used anywhere in our life. I hope this project could also demonstrate my knowledge on geospatial data processing.

6. **Conceptual design**:



7. **User interaction**:


User login
password correct?
No
Please retry
Yes
User main page
My friends list
My trips list
For each trip
Create/Read/Update/Delete cousines
Uplate/Delete photos
Search a friend via mail address
No
Yes
Delete the friend
Create my new trip
Delete my existing trip
Update my existing trip
Please retry
Add the friend in my friends list

Final Report

1. **README**

   Please strictly follow the steps below.

   1. STEP1

      Import the self-contained dump file into the local database.

   2. STEP2

      Install all the prerequisites.

      Python3.x && pip

      pip install Django

      pip install pyjwt

      NOTE that Django uses mysqlclient as driver, we just assume you already have mysqlclient installed on your computer. If not, please use pip install mysqlclient

   3. STEP3(OPTIONAL)

      Change the ROOT/mysite/setting.py file in the project.

      From Line 82 to 86, please change the content to the config of your local database like 127.0.0.1:3306, etc.

      NOTE that you don’t have to change the setting here, because the database I use is a remote server with a public IP, which you could get access to anywhere.

   4. STEP4

      RUN the server by:

```
python manage.py runserver [0.0.0.0:8000]
```

NOTE that you could customize the bind IP and PORT, by default, the server is binding on 127.0.0.1:8000. We suppose that you use the default IP and PORT from here.

Hopefully, you will see the log in the console saying:

```
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

Now, our server is on! We could go to the browser for further exploration!

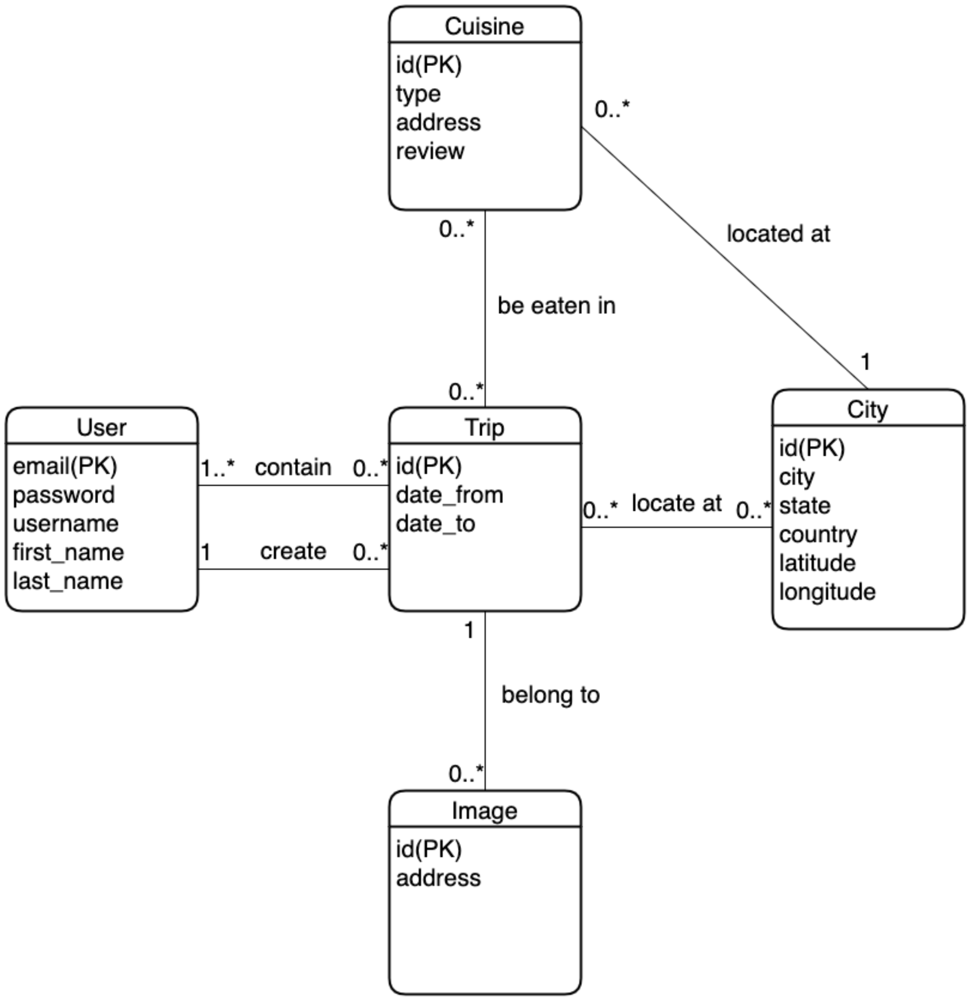
5. STEP5

   Go to http://127.0.0.1:8000/

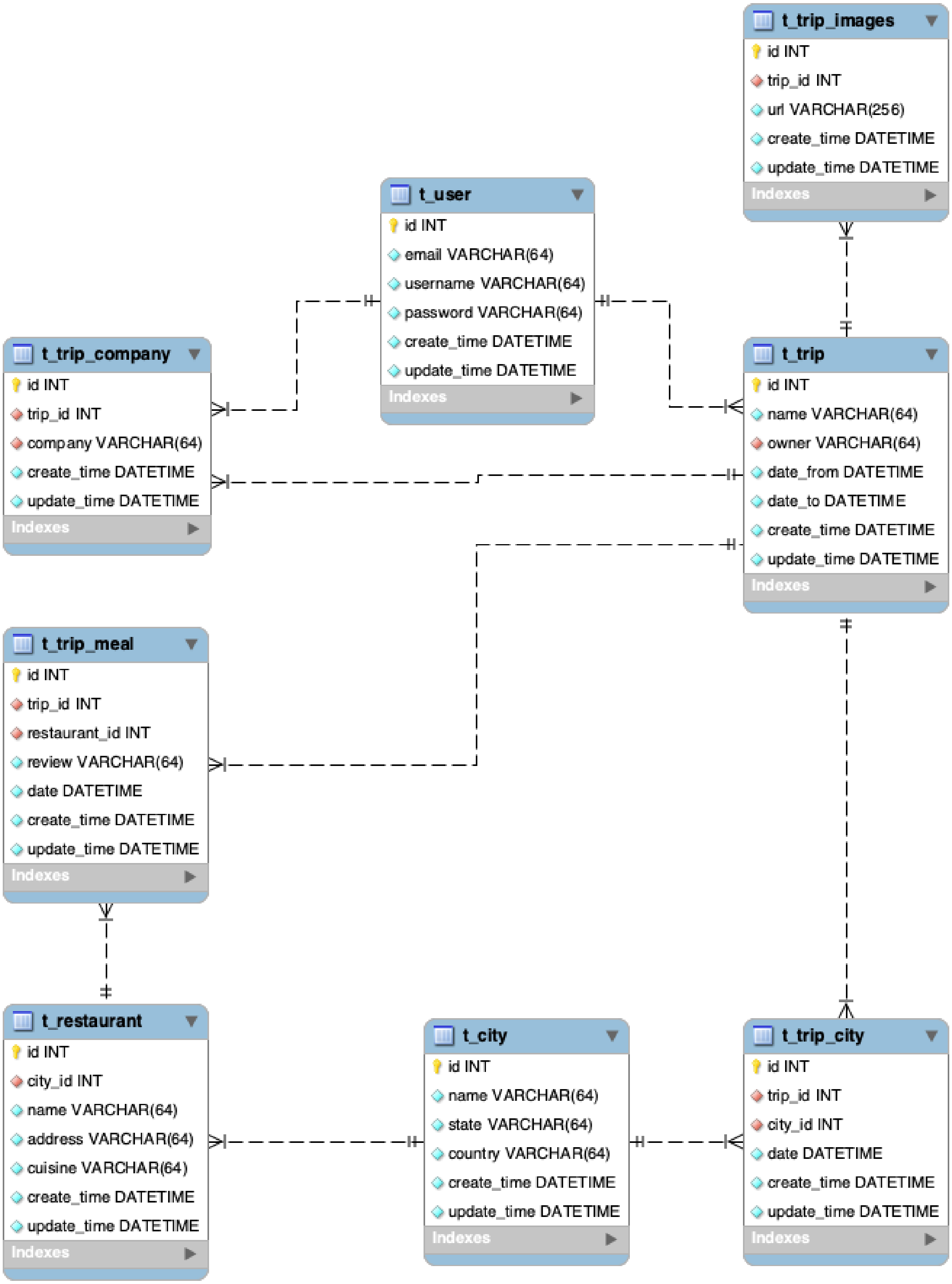
6. STEP6

   Login page: http://127.0.0.1:8000/, after login, you will be directed to the trips page http://localhost:8000/trips. If you want to logout and change an account, please go back to the Login page again and relogin! After clicking the details button after each trip, you will be directed to http://localhost:8000/detail/<id> page. To add itinerary or meals, please first refer to the http://localhost:8000/cities page and add the cities and restaurants you would like to add. Once you add these cities, you can find the item in the input box.

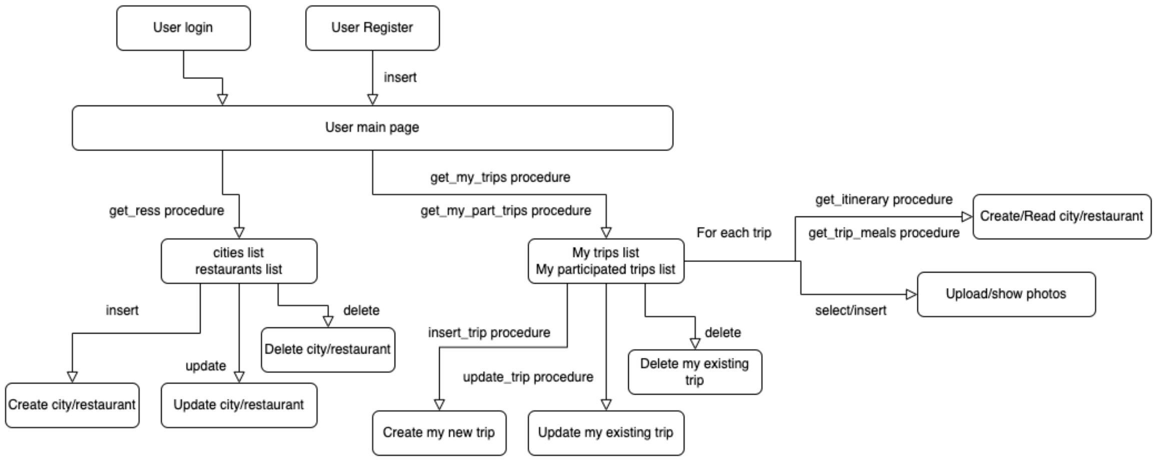
2. **Final UML**



3. Final Logical Design


t_trip_images
id INT
trip_id INT
url VARCHAR(256)
create_time DATETIME
update_time DATETIME
Indexes
t_user
id INT
email VARCHAR(64)
username VARCHAR(64)
password VARCHAR(64)
create_time DATETIME
update_time DATETIME
Indexes
t_trip_company
id INT
trip_id INT
company VARCHAR(64)
create_time DATETIME
update_time DATETIME
Indexes
t_trip
id INT
name VARCHAR(64)
owner VARCHAR(64)
date_from DATETIME
date_to DATETIME
create_time DATETIME
update_time DATETIME
Indexes
t_trip_meal
id INT
trip_id INT
restaurant_id INT
review VARCHAR(64)
date DATETIME
create_time DATETIME
update_time DATETIME
Indexes
t_restaurant
id INT
city_id INT
name VARCHAR(64)
address VARCHAR(64)
cuisine VARCHAR(64)
create_time DATETIME
update_time DATETIME
Indexes
t_city
id INT
name VARCHAR(64)
state VARCHAR(64)
country VARCHAR(64)
create_time DATETIME
update_time DATETIME
Indexes
t_trip_city
id INT
trip_id INT
city_id INT
date DATETIME
create_time DATETIME
update_time DATETIME
Indexes

4. Final User Flow



5. Lessons Learned

1. Technical expertise gained

   Although I have 2 years of working experience of software and database, I have also learned a lot in this project. Before, I rarely used procedures, functions and triggers, however, after the project, I realized the convenience of all the objects. It's also my first time to write so much front end html, css and javascript code as well as the django framework. I encountered some technical issues during the process like the image uploading, but I am glad I solved them by searching for a large amount of documents. Plus this is a one-person group project, so I'm glad I established a whole system from 0 to end by myself.

2. Insights, time management insights, data domain insights etc.

   We should definitely start this kind of big project early. The more time you have, the more perfect your project will be. At the end of the deadline, because I didn't have much time, I gave up some optimization on the front end. Insteadly, I focused more on the database itself like the procedures. If I had more time, the website would be more fancy.

For the data domain, we could add more interesting attributes of the trip, like the review, notification or other things. We could mention that in future work.

3. Realized or contemplated alternative design / approaches to the project

   We could change MySQL to MongoDB to store some unstructured data, like the geospatial data of the city and the routine we have made. If we change to MangoDB, we will have more flexibility but the working of join will also be a little more demanding.

   For the framework, we could also try Flask, Java with Spring or maybe Golang, which is such a popular web development language.

4. Document any code not working in this section

   For the image upload code, we just read the static file (images) from the project directory. If we make the website to the production mode, we need to have some object storage like Amazon S3 to store all the images and return a url link to the database so that the front end code could easily get the image resource. This is also a great topic of image or object storage.

6. Future work

1. Planned uses of the database

   As I mentioned in the proposal, the database could be used as the storage of all the trips that could be shared among all the friends. We could make the website be like the application Zenly to share all the trips among a group. Additionally, we could also share locations, photos, etc.

2. Potential areas for added functionality

We could make the website a mobile application and make it a socialization platform. Although I didn’t develop the add friends function in my project, we have a user system that could be used as the social platform. And we also have the “participated trips” which connect the friends together!

3. No future uses or work can be documented if justification is provided.

   We still have potential to make our database conceptual design and the front end design better. Hope we have energy to make our system better after the justification!