Simon Yu
Final Report
4/17/2024

README with Technical Specifications:
PROGRAMMING LANGUAGE : PYTHON
USING SQL WORKBENCH
How to run the application:
1. Make sure to have both pymysql and tabulate dependencies installed by doing these commands on the command line:
pip install pymysql (for mysql workbench)
pip install tabulate (for reading data output table later)
2. Once dependencies are installed, run in the command line "python .\Project_Yu.py" to run the code that was downloaded. (Project_Yu.py is the name of the python file)
3. The username and password are prompted and the user has to connect to the database through the mysql username and password. (Use your own!)
4. Once the connection is there, a day-to-day basis of the laundromat operation is outputted. The first is a scenario of a customer coming in. You are to get their information such as their name, age, etc. You are also going to get the color of their laundry bag. This is usually done because the laundry bag is needed to put all the folded clothes in (known through prior real life experience working in a laundromat).
5. A prompt (Create operation) for the customer's phone number and other information is needed for the ticket creation. Once the ticket is created, a list of all the tickets that have been done throughout the day so far is displayed.
6. The next scenario is for the Update operation where if a user is to change their pick-up date, you will be asked for their ticket number and the date they want to change it to.
7. Another scenario is for the Delete operation where the customer wants a refund and so their ticket should be deleted from the database table.
8. Finally, an output is made with the Read operation. We are reading through all the tickets that have been created today and specifically looking at the cost of each ticket so that we can get the total earnings of the day.
9. Connection is then closed once all steps of the application are run.
NOTE: error exceptions are made for each function in the python code so that if there are any errors, the exceptions will take effect.

Textual Description:

A customer has a unique ID, name, age, and customer level (new, old). A customer can drop one or many laundry bags. The laundry bag has its own unique id and the color of the bag. A ticket is created that serves as a recipe for the customer and a ticket for the laundromat.

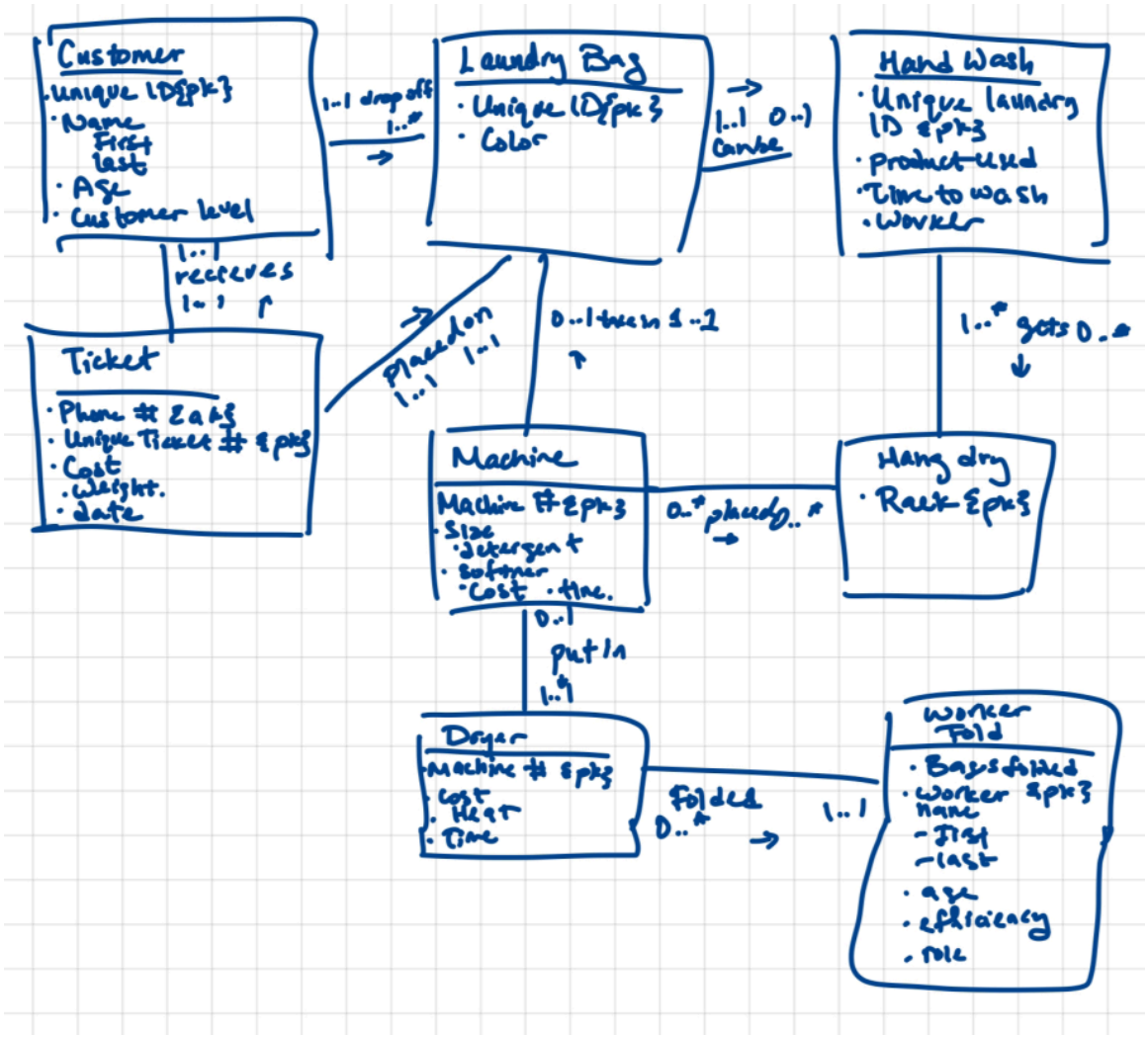
The ticket is placed on the laundry bag as reference and consists of: the customer's phone number, the unique ticket number, the total cost, the pick-up date, the weight of the laundry, and both a washing and a drying method. Only one ticket per laundry bag. The customer also gets one recipe which is the copy of the ticket.

A bag of clothes can be washed in two different ways. It can be done through the washing machine or through hand wash. If it is through the washing machine, it will be dedicated to a unique number machine. The washing machine consists of its id, the size either regular or large, both the detergent type and the softener type, the total time it takes for the wash, and the laundry bag it is taking in. If it is through the hand wash, it just has the product that is used to wash the clothes and how long of a wash it is. A washing machine can only wash one laundry bag at a time and a laundry bag can only go into one washing machine. Same goes with hand-wash.

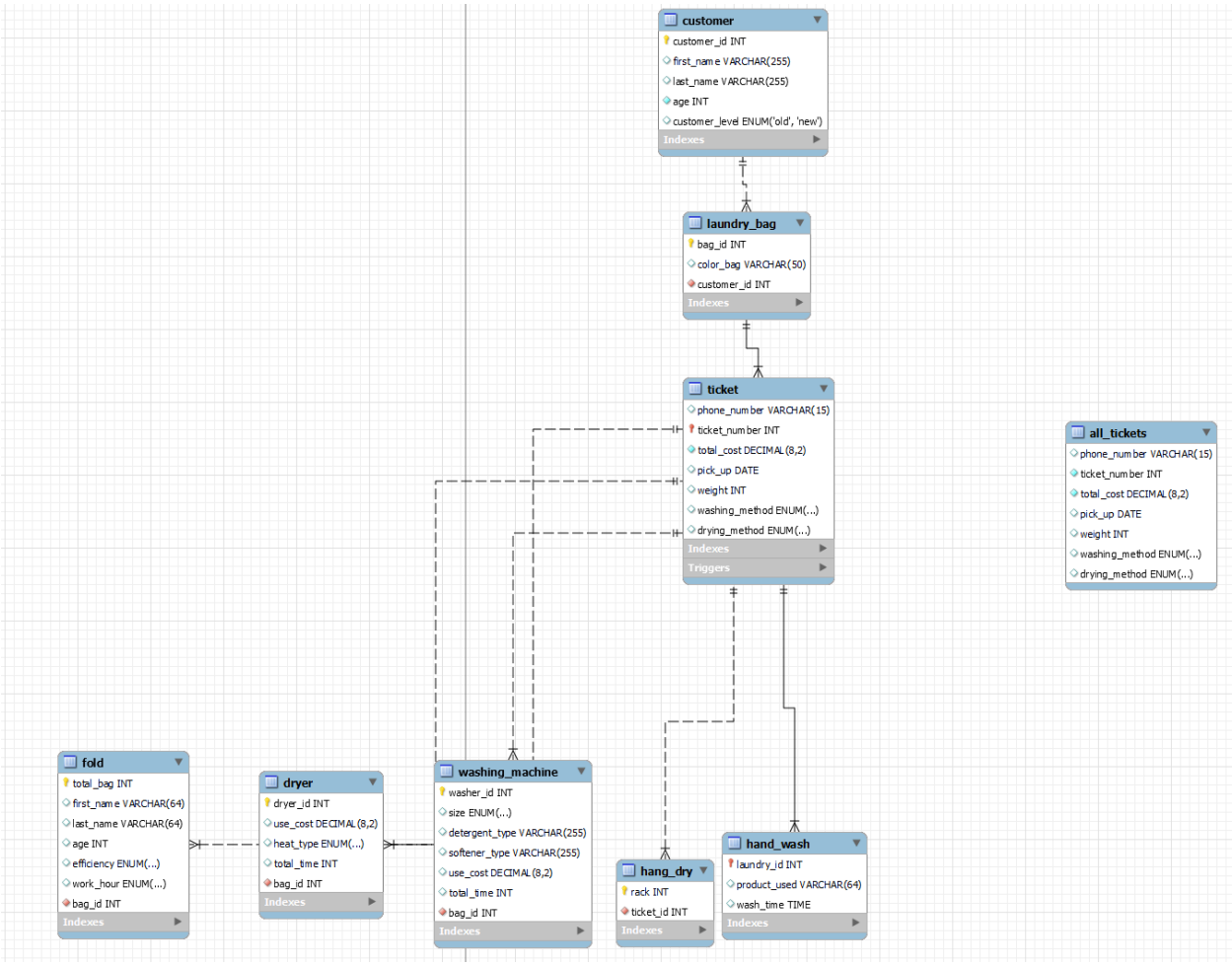
A bag of clothes can also be dried in two different ways: a dryer or hang dry. If it is dried through a dryer, it also has to be folded. A dryer consists of the dryer id, the cost of using it, the heat type, the time it takes default at an hour, and the laundry it is taking in. If the laundry is dried through hang dry, it just needs to know the rack that the clothes are being hung on. Either clothes from hand wash or washing machine can go to the drying rack, but hand washed clothes can only go to the rack and not the dryer. The dryer can only receive one laundry bag at a time.

If the clothes are to be folded, it will be folded by a worker. The fold consists of the laundry bags that are being folded, and the worker. The worker is made up of their: first name, last name, age, efficiency (slow, normal, fast), the type of worker (part-time, full-time), and the laundry bag they are folding. Clothes will be folded by one worker but a worker can fold multiple laundry bags throughout the day.

UML DIAGRAM:


Customer
Unique ID {pk}
Name
First
last
Age
Customer level
1..1 drop off 1..*
Laundry Bag
Unique ID {pk}
Color
1..1 0..1 Can be
Hand Wash
Unique laundry ID {pk}
Product used
Time to wash
Worker
1..1 receives 1..1
Ticket
Phone # {alt}
Unique Ticket # {pk}
Cost
Weight
date
placed on 1..1 1..1
0..1 turns to 1..1
1..* gets 0..*
Machine
Machine # {pk}
Size
detergent
softener
Cost time
0..* placed 0..*
Hang dry
Rack {pk}
0..1 put in 1..1
Dryer
Machine # {pk}
cost
Heat
Time
folded 0..* 1..1
Worker Fold
Bags folded
Worker {pk}
name
-first
-last
age
efficiency
role

LOGICAL DIAGRAM:



FINAL USER FLOWS:

- Make sure to follow the templates of the way things are written and ask as prompts for the program to run successfully, or else you will bump into an exception, closing the program.
- Once the program is executed, the prompts are meant to be followed to get a full comprehensive look at the CRUD operations at work, each at least used once.
- CREATE : customer, laundry bag, ticket, dryer, washing machine
- READ: tickets, ticket's costs, made into a read of total earning
- UPDATE: updating pick-up date to any date the client wants
- DELETE: deleting a ticket when a customer wants a refund, for example

All code runs, some functions and procedures/triggers may not have been implemented because they encountered bugs and were purposely left out and may be used in future works when figuring out how to use them!

LESSON LEARNED:

Some things I learned from this project is how interesting it was creating my own database and how difficult it can be. I had a hard time trying to get the foreign keys to work with one another and how different tables were to be connected to each other the way I wanted them to be. For example, I didn't want the bag id of most of my tables as the foreign key because most of the information was based off the ticket and so I wanted the foreign keys to be referencing the ticket that was what is attached to the laundry bag. Time management could have been done better as I had to spend a lot of time creating different functions, procedures, and triggers as well as testing them to see that they worked the way that was intended. Another approach to this project would definitely be incorporating different workers and having them specified for different jobs like in the real world where there are workers working as the receptionist and there are those who have the muscles to do all the washing and drying, and those who are there for just folding clothes and facilitating the laundromat. I felt like once I was able to create the procedures and functions that I wanted, I was able to easily used them in the front-end python program by just "callproc" them and use them the way I wanted to which for this project was through prompts and using those to create, read, update, and delete certain tuples and objects.

FUTURE WORKS:

A future work that I can already see me building is updating the ticket orders. Since I was working by myself, it was hard to incorporate more designs that I wanted. One design was that if the weight of the bag was more than 27 pounds, I was thinking of updating the ticket order so that it updated the total_cost to be added an extra $1.50. Additionally, there are times where customers want to use their own special detergent or softener and so that should be changed when they go into the washing machine. Since my parents own a laundromat and I used to work there, with my prior experience, I can come up with more sophisticated triggers, functions, or procedures that are seen from different unexpected scenarios and ultimately, this program can be used to help store all the customers, laundry bags, and tickets for logging and also for their earnings of the day. It will be helpful to log these information as a database because I know my parents like to log these information down but they are doing it through a notebook using a pen and ruler.