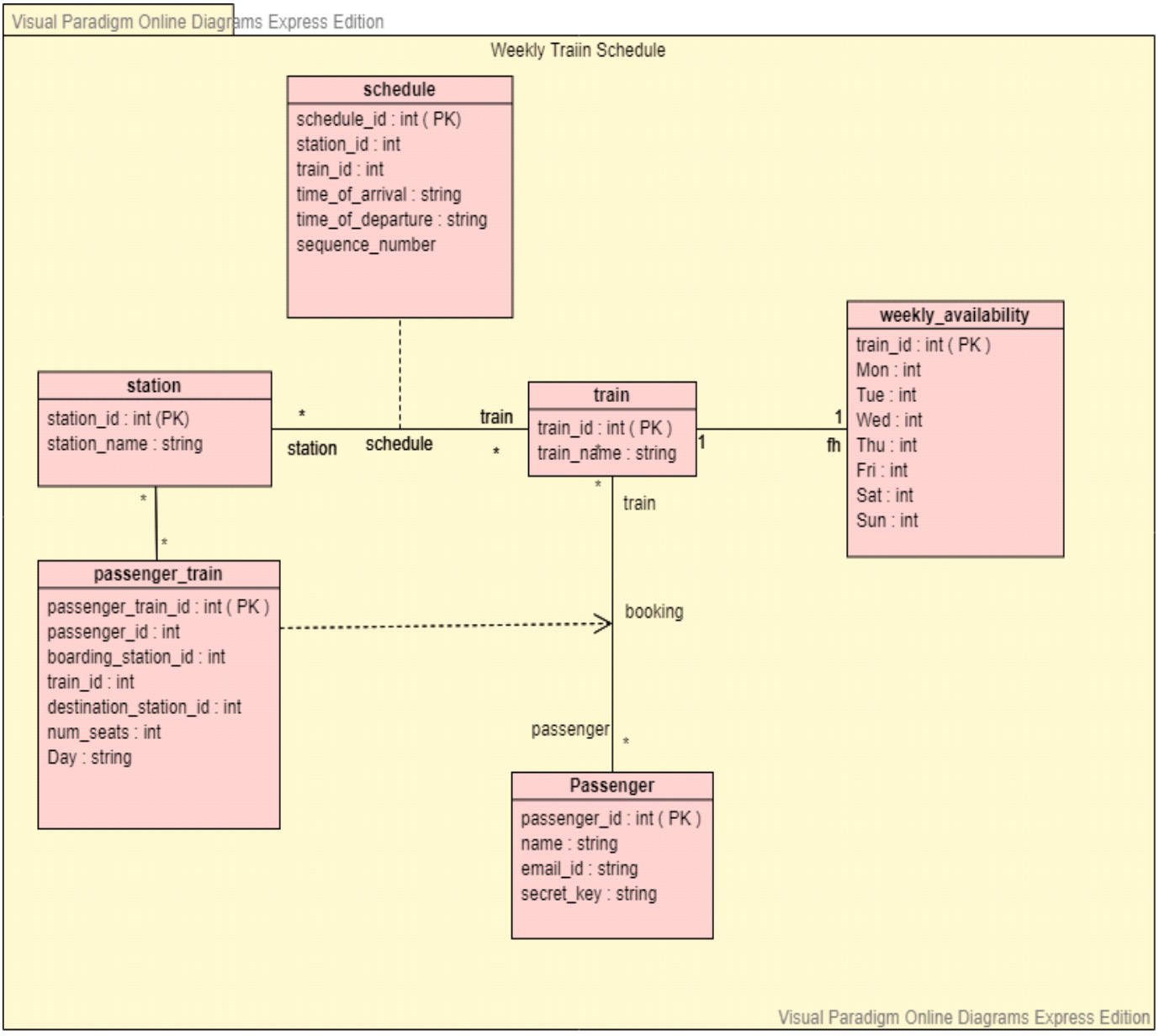
Project Report

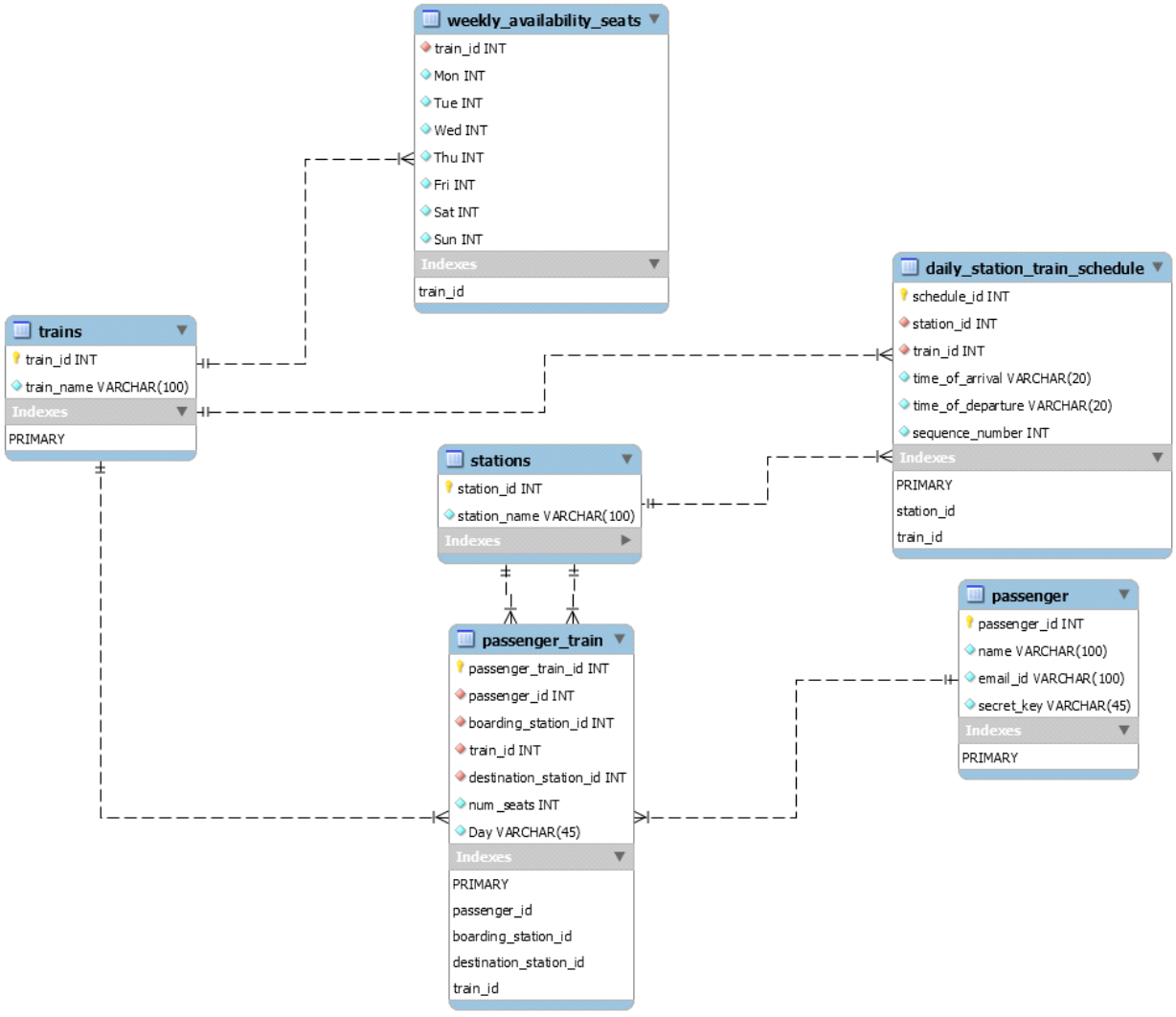
Saturday, June 20, 2020 11:49 PM

1 . UML



Please find the UML notation devised using Visual Paradigm Online Diagrams Express Edition Tool.

2 . Logical Design :



3 . Lesson Learnt and Future Work Section :

Lesson Learnt :

1 . **Host Language Benefits :** Since I come from Firmware background usually dealing with memory and MIPS . Using an application oriented like Python for first time was a wonderful experience . Owing to less knowledge of the same I spent a lot of time trying to write the code from scratch for tabular display of data and other modules , which already existed with this language.I regret spending time on something that has been already implemented but a good lesson learnt to know more about the giveaways with the choice of a language.

2 . **Spending more time with Design** : I spent a good quantity as well as quality time in logical design , but from the proposal to final report I still made many changes to it (adding extensive features , normalizing the tables) . Which makes me wonder that wish I would have put some more time and take an expert opinion in office hours to come up with a good project proposal therein to avoid massive design changes post proposal.

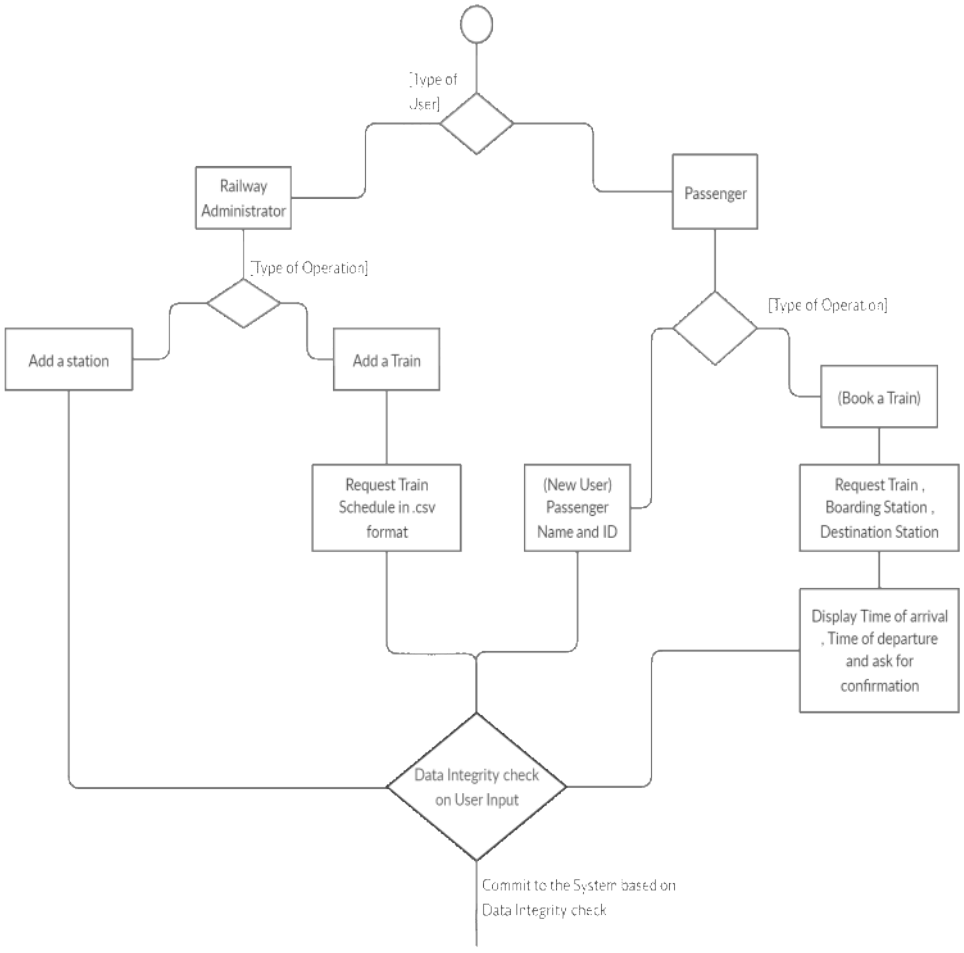
Future Work Section :

1 . Scaling it to yearly schedule : I have designed a railway tickets and services portal only for a weekly basis , but design is scalable enough to make it a yearly based service by adding one more table.

2 . Pricing of tickets : Currently there is no pricing of tickets , but we can easily extend the database to include pricing of tickets and adding a functionality in admin object to set a price for a ticket of a train

3 . RDBMS vs NoSQL : Although I selected RDBMS for this project , but I think in future work a detailed analysis on why RDBMS vs NoSQL and which could be better for this database would further improve the performance of the application.

4 . User Activity :



User activity follows above call-flow in application as well .

Administrator :

He \ She is provided with features to Add a Station and Train .
Adding a train would expect him \ her to provide the schedule in a format that the application would understand which goes as below :

<City>,<Arrival Time>,<Departure Time>,<Sequence of station>

Mumbai,13:00 ,13:10 ,1
Pune ,18:30 ,19:10 ,2
Kolkata ,21:30 ,21:40 ,3

Above train makes a journey from **Mumbai** to **Kolkata** with a stop at Pune in between

Passenger :

He \ She is provided with features to
1 . Register as a new user
2 . Book a ticket
3 . Display Tickets
4 . Cancel Ticket

For registration , his email-id is used a primary key but eventually there won't be any email acknowledgement that would be used for confirmation of email-id as it doesn’t come in the ballpark of the course .
Application is quite user interactive , with each prompt giving a detail walkthrough of it's functionality.

5 . README or Technical Description :

Tools used : MySQL Workbench 8.0.20 Community Edition for database design
Notepad++ For editing python code

Software :
Python : Version 3.8.3
Modules : mysql.connector
mysql
csv
tabulate

PIP : Version 20.1.1

Installation steps : pip install tabulate
pip install csv
pip install mysql-connector-python
pip install mysql

Database : RDBMS with MySQL used for querying the database

Code Hierarchy :

/<main_dir> :
__init__.py
appcode.py
/<main_dir>/<userdefs>
__init__.py
admin.py
user.py

Run Command :
python appcode.py

(It will prompt for database credentials)

Admin credentials : username : root , passkey : root