



Kids Crew

Sharayu Thosar & Pranjali
Bhatt

Group 11

Problem Definition



Kids Crew is an all-in-one solution for childcare. The company provides families with children care services of babysitting and pick up drop off for the 1–10-year age group.

Individuals looking for part time jobs can register themselves under the category of service they are interested in working. Parents can find a perfect fit based on their budget and requirements.

Aim: To create a database model for the company and gain insights into how to enhance the quality of service.

Problem Definition



- Babysitters seeking job opportunities can register with their availability, age, bio, bank account details, type of services (babysitting, pickup -drop off).
- Parents can create an account on our application. The app will collect details. A user ID is then generated and assigned for every account.
- A separate children's database is maintained
- Once the account is created the parents can choose and book a service for as many children as they would like as per their requirement. They need to book one day prior by checking the availability of the sitter. After the service is completed, parents pay the sitter through the app and can give a rating.
- A parent can rate a sitter between 1-5 only after the service has been provided. The experience of each babysitter is stored on a table that includes their ID, number of hours logged for infants (1-3), toddlers (4-6), children (7-10) and rating.
- The initial rate (no experience) for babysitting will be \$15. With experience the rate (hourly pay) increases.

Requirements

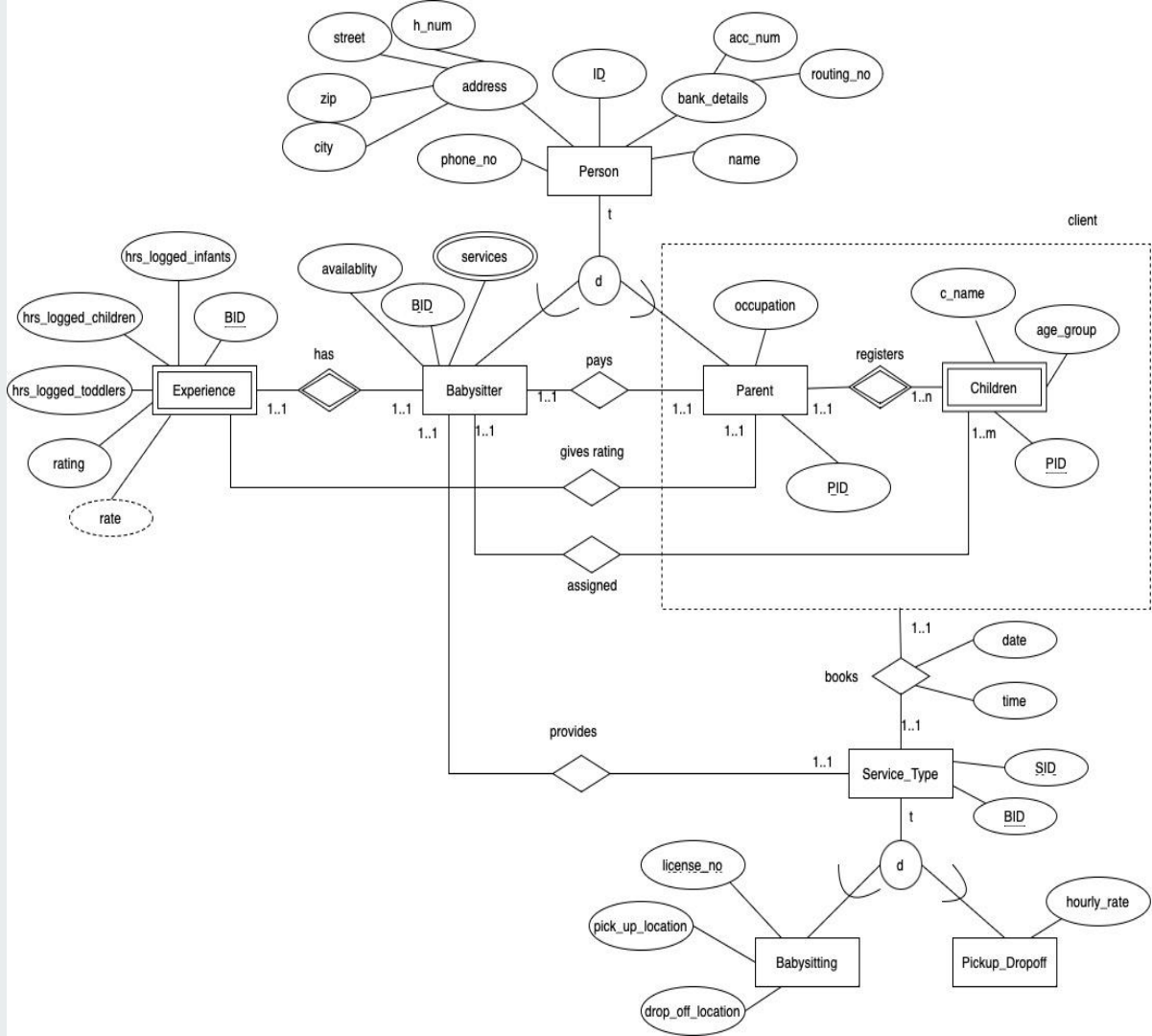


Tools and softwares used for the project implementation

- UML and ER Model: draw.io
- Data Population: mockaroo.com
- Database: MySQL Workbench
- Application: Python (Jupyter Notebook)
- NoSQL: Neo4j

Conceptual design

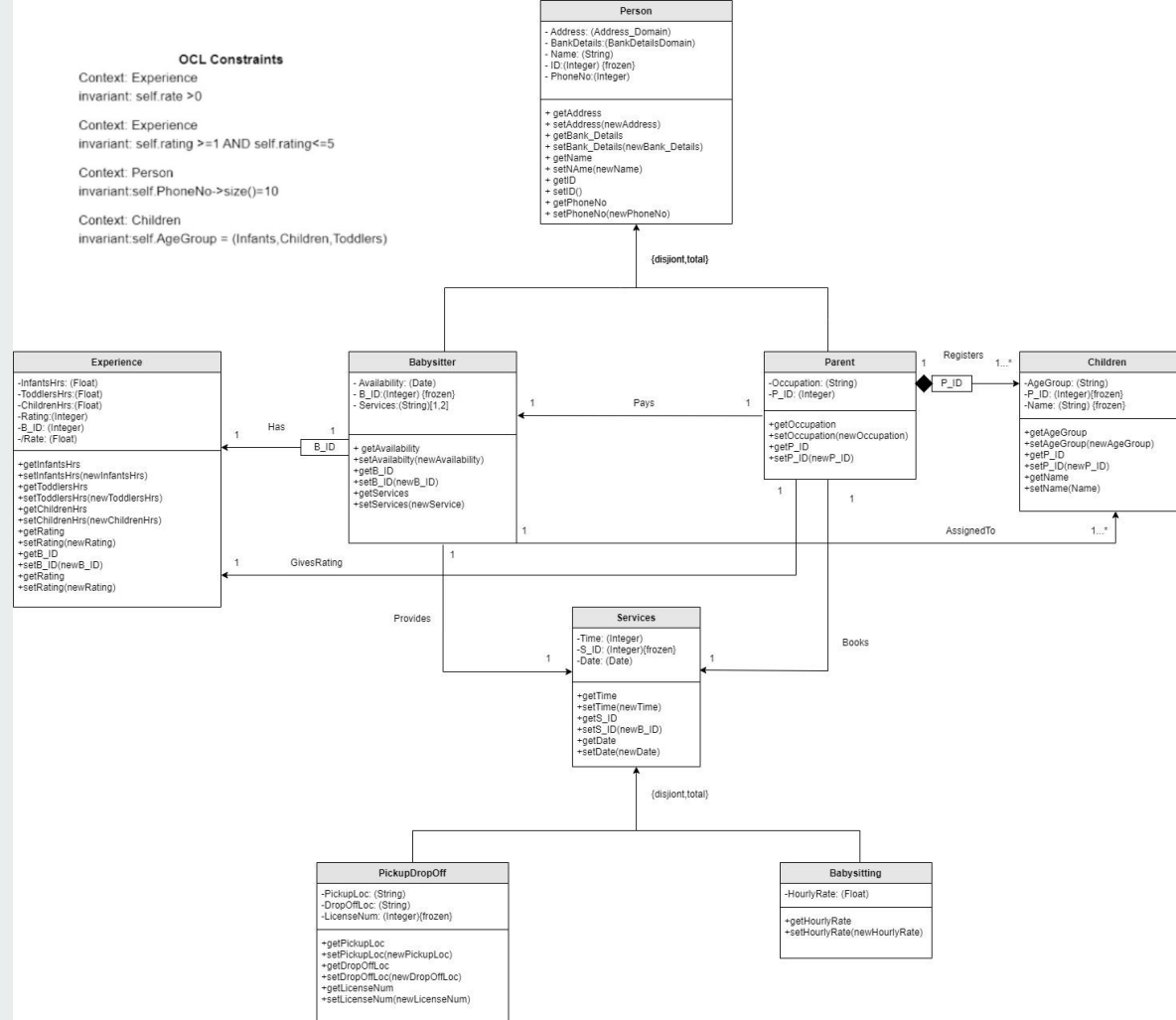
EER model



Conceptual design

UML Class Diagram

[Link to Diagram](#)



Relational model

Parent(PID, occupation, acc_num, p_name, phone_no, HNum, street,zip,city,routing_num, acc_num)

- PID is the primary key

Babysitter(BID,acc_num, name, phone_no, HNum, street,zip,city, availability,PID,routing_num, acc_num)

- BID is primary key
- Acc_num NOT NULL
- PID is foreign key referring to PID in Parent for the relation Pays;NOT NULL

Experience(BID, hrs_logged_infants, hrs_logged_toddlers, hrs_logged_children, rating, rate, PID)

- PID is a foreign key referring to PID in Parent for relation gives rating;NOT NULL
- Experience is a weak entity. It refers to the BID in Babysitter. BID is the primary and foreign key

Client(PID,BID, SID, Name)

- PID, Name is primary key as it comes under aggregation of Parent and Children
- BID is a foreign key referring to BID in Babysitter;NOT NULL
- SID is foreign key referencing to SID in ServiceType as client books relation;NOT NULL

PickupDropOff(BID, license_no, pickup_loc, drop_off_loc)

- BID is the primary key
- BID is a foreign key referring to BID in Babysitter;NOT NULL
- License_no, BID are candidate keys

Book(SID, date, time, BID)

- SID is foreign key referencing to SID ;NOT NULL
- BID is a foreign key referring to BID in Babysitter.;NOT NULL
- BID,SID is primary key

Babysitting(hourly_rate,BID)

- BID is primary key
- BID is a foreign key referring to BID in Babysitter;NOT NULL

Parent(PID, occupation, acc_num, name, phone_no, HNum, street,zip,city)

- PID is the primary key

Children(Name, age_group, PID,BID)

- PID and Name is the primary key
- Children Entity is a weak entity. PID is a foreign key referring to Parent for registers relation;NOT NULL
- Name is included as a parent can have multiple children
- BID is foreign key referring to BID in Babysitter for assigned relation;NOT NULL

ServiceType(BID,SID, Service)

- BID is a foreign key referring to BID in Babysitter ;NOT NULL
- SID, BID is the primary key



Next

MySQL

NoSQL

Application