Daycare Management System

SHAEER MUSARRAT SWAPNIL
ROLL: 2007116

CSE3110: Database Systems Laboratory

Introduction

Project Overview:

The daycare management system database effectively organizes the management of daycare operations crucial for parents and caregivers, featuring tables like DaycarePackages, Caregiver, Parent, Child, and DailySchedule. It tracks caregiver details, parental information, and child profiles, primarily aiding in the comprehensive care and supervision of children. The Child table links caregivers and parents, ensuring personalized attention. The DailySchedule table manages the daily activities and timing for each child, connected to specific daycare packages, which helps in planning and resource allocation. Overall, this system supports daycare facilities in maintaining structured care delivery and managing operations efficiently through detailed data management and association between different entities involved.

Importance of Database Management in Daycare:

Database management plays a pivotal role in daycare operations by ensuring operational efficiency and regulatory compliance. The structured tables like DaycarePackages, Caregiver, Parent, Child, and DailySchedule facilitate accurate tracking of caregiver details, parental information, child profiles, and daily activities. This setup is crucial for maintaining high standards of child care, safety, and personalized attention. By providing a system that aids in resource allocation and scheduling, database management helps optimize staff assignments and improve the quality of care provided to each child. Additionally, efficient data management supports strategic decision-making for daycare centers, enhancing overall service quality and parent satisfaction.

Objectives:

1. **Child Records Management**: To maintain detailed, up-to-date records of each child, including personal information, health data, and daily activities, to ensure personalized care and safety.
2. **Caregiver Scheduling and Management**: To effectively manage caregiver assignments and schedules, ensuring each child receives appropriate attention and care based on their specific needs and the qualifications of the caregivers.
3. **Parent Interaction Enhancement**: To facilitate seamless communication between parents and daycare staff via updates, reports, and direct messaging, thereby improving transparency and trust.
4. **Operational Efficiency in Child Care**: To optimize the daycare operations through streamlined processes for attendance, scheduling, and monitoring of daily activities, enhancing the overall efficiency and quality of care.
5. **Attendance and Time Tracking**: To implement accurate attendance tracking and time management for children, enabling precise monitoring of check-ins, check-outs, and overall time spent at the daycare.
6. **Emergency Response Management**: To develop protocols and systems for managing emergencies, including quick access to child medical information and contact details, ensuring rapid and effective responses to health or safety issues.
7. **Flexible and Scalable System Architecture**: To build a scalable and flexible database system that can adapt to increasing numbers of children and evolving operational needs, ensuring long-term sustainability and the ability to integrate new functionalities as required.
8. **Regulatory Compliance and Reporting**: To ensure the system complies with local and national regulations regarding child care operations, including data privacy laws, and to provide necessary reporting features for administrative and compliance purposes.

These objectives aim to create a robust and efficient daycare management system that supports excellent child care, streamlines administrative processes, and enhances communication between the daycare staff and parents.

Database Schema:

The daycare management system database organizes critical information that ensures smooth operation and high-quality childcare services. Below is a detailed overview of each table in the database schema:

✅ **DaycarePackages Table: describes different daycare packages available.**

- **Attributes:**
    - package_id (primary key)
    - package_name
    - description
    - cost

✅ **Caregiver Table: Stores information about the caregivers responsible for the children.**

- **Attributes:**
    - caregiver_id (primary key)
    - caregiver_name
    - caregiver_email
    - caregiver_phoneno
    - caregiver_experience

✅ **Parent Table: Stores information about the parents of the enrolled child.**

- **Attributes:**
    - parent_id (primary key)
    - father's_name
    - mother's_name
    - father's_email
    - mother's_eemail

- father's_phoneno
- mother's_phoneno

✅ **Child Table: Stores information about each child enrolled in the daycare.**

- **Attributes:**
  - child_id (primary key)
  - name
  - date_of_birth
  - gender
  - address
  - parent_id (foreign key referencing Parent table)
  - caregiver_id (foreign key referencing Caregiver table)

✅ **DailySchedule Table: Specifies the daily schedule for each child, including the daycare package they are enrolled in.**

- **Attributes:**
  - schedule_id (primary key)
  - child_id (foreign key referencing Child table)
  - start_time
  - end_time
  - package_id (foreign key referencing DaycarePackages table)

This schema captures all necessary details for managing daycare operations effectively, from caregiver assignments to the daily activities of children, facilitating thorough oversight and structured care.

Table Relationships:

```
-- Creating the DaycarePackages table
CREATE TABLE DaycarePackages (
    package_id NUMBER PRIMARY KEY,
    package_name VARCHAR2(100),
    package_description VARCHAR2(255),
    package_cost NUMBER(20)
);

-- Creating the Caregiver table
CREATE TABLE Caregiver (
    caregiver_id NUMBER PRIMARY KEY,
    caregiver_name VARCHAR2(100),
    caregiver_email VARCHAR2(100),
    caregiver_phoneno NUMBER(20)
    caregiver_experience VARCHAR2(100),
);

-- Creating the Parent table
CREATE TABLE Parent (
    parent_id NUMBER PRIMARY KEY,
    parent_name VARCHAR2(100),
    parent_phoneno NUMBER(20),
    parent_email VARCHAR2(100)
);

-- Creating the Child table
CREATE TABLE Child (
    id NUMBER PRIMARY KEY,
    name VARCHAR2(100),
    date_of_birth DATE,
    gender VARCHAR(20),
    address VARCHAR2(255),
    parent_id NUMBER(20),
    caregiver_id NUMBER(20),
    FOREIGN KEY (parent_id) REFERENCES Parent(parent_id),
    FOREIGN KEY (caregiver_id) REFERENCES Caregiver(caregiver_id)
);

-- Creating the DailySchedule table
CREATE TABLE DailySchedule (
    schedule_id NUMBER(20) PRIMARY KEY,
    child_id NUMBER(20),
    start_time TIMESTAMP,
    end_time TIMESTAMP,
    package_id NUMBER,
    FOREIGN KEY (child_id) REFERENCES Child(id),
    FOREIGN KEY (package_id) REFERENCES DaycarePackages(package_id)
);
```

Schema Diagram:

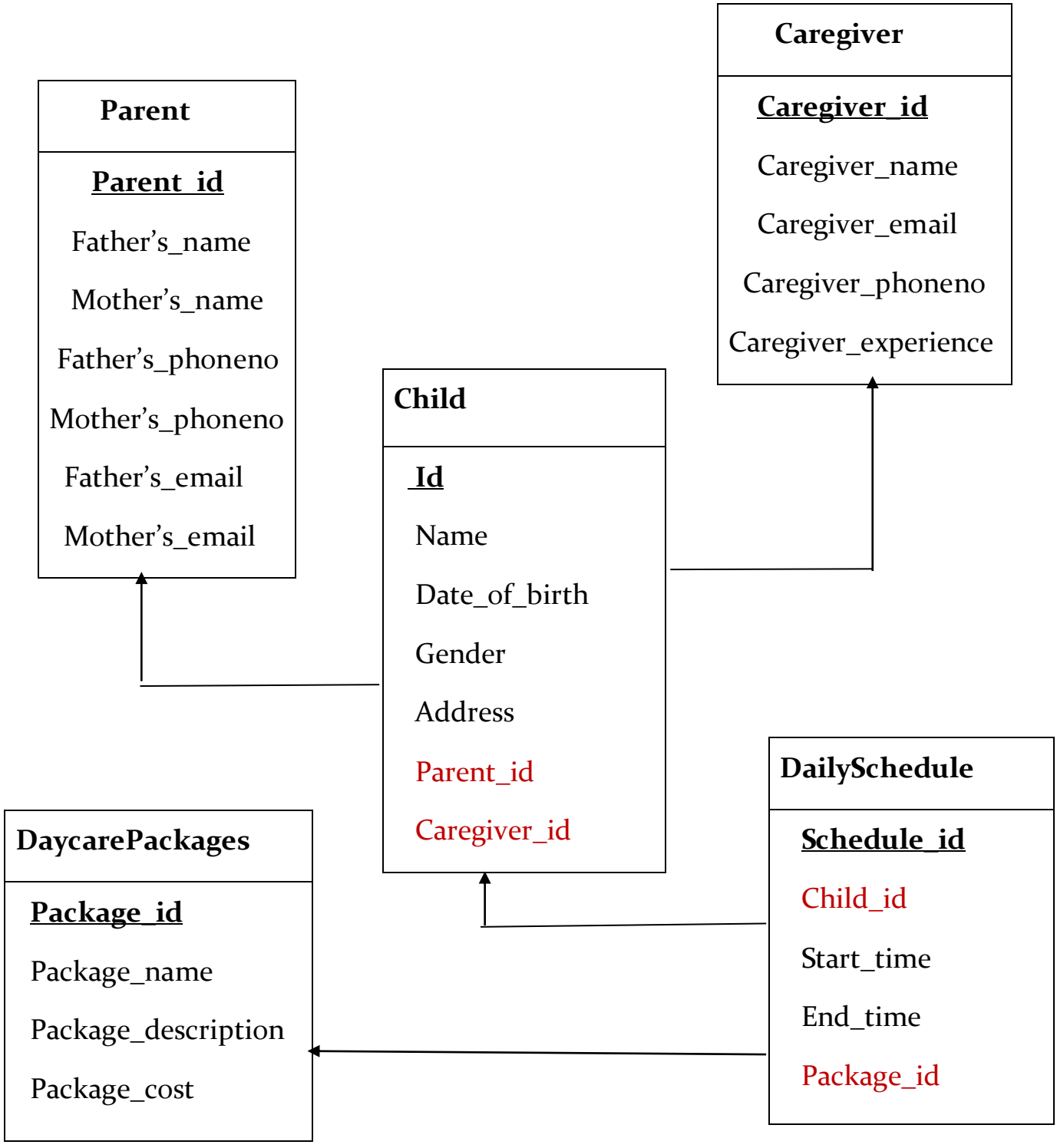

Figure: Schema diagram of Daycare Management System

E-R Diagram:

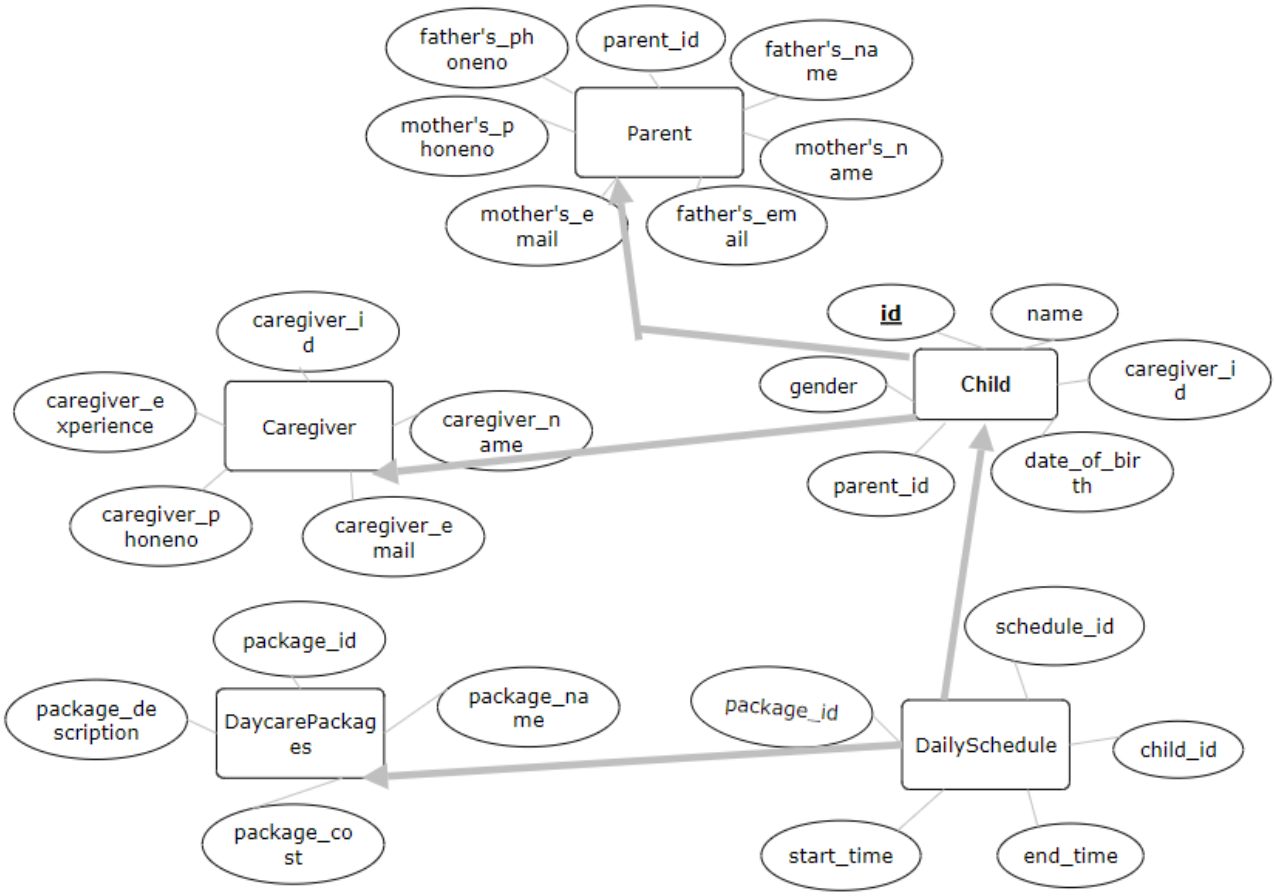

Figure: E-R diagram of Daycare Management System

SQL Queries:

To facilitate a wide range of informational needs for users, several SQL queries have been designed to extract useful information from the database. Here are some example queries:

1. Retrieving Daycare Package Information by Cost:

```
SELECT * FROM DaycarePackages WHERE package_cost = 150.00;
```

2. Selecting Children Assigned to a Specific Caregiver:

```
SELECT * FROM Child WHERE caregiver_id = 2;
```

3. Retrieving Children Enrolled in a Specific Package:

```
SELECT * FROM Child WHERE id IN (SELECT child_id FROM DailySchedule
WHERE package_id = (SELECT package_id FROM DaycarePackages WHERE
package_name = 'Premium Care'));
```

4. Updating Daycare Package Names:

```
UPDATE DaycarePackages SET package_name = 'Standard Care' WHERE
package_name = 'Basic';
```

5. Using a UNION to Retrieve Package Names:

```
SELECT package_name FROM DaycarePackages WHERE package_name LIKE 'B%'
UNION SELECT package_name FROM DaycarePackages WHERE package_name
LIKE '%Care%';
```

6. Using WITH Clause for Max Package Cost:

```
WITH MaxCost(cost) AS (SELECT MAX(package_cost) FROM DaycarePackages)
SELECT * FROM DaycarePackages, MaxCost WHERE
DaycarePackages.package_cost = MaxCost.cost;
```

7. Counting Distinct Daycare Package Names:

```
SELECT COUNT(DISTINCT package_name) AS number_of_packages FROM
DaycarePackages;
```

8. Average Number of Children per Package:

SELECT package_id, AVG(child_count) AS avg_children FROM (SELECT package_id, COUNT(*) AS child_count FROM DailySchedule GROUP BY package_id, child_id) GROUP BY package_id;

These queries demonstrate a variety of SQL operations including selection, aggregation, updates, and the use of advanced SQL clauses like WITH and UNION, tailored to manage and analyze a daycare center's data efficiently.

PL/SQL:

1.Fetch and Display All Caregivers and Their Corresponding Package Names Using a Cursor:

```
DECLARE
   CURSOR c_caregivers_packages IS
      SELECT cg.caregiver_id, cg.caregiver_name, dp.package_name
      FROM Caregiver cg
      JOIN Child ch ON cg.caregiver_id = ch.caregiver_id
      JOIN DailySchedule ds ON ch.id = ds.child_id
      JOIN DaycarePackages dp ON ds.package_id = dp.package_id;
   v_caregiver_id Caregiver.caregiver_id%TYPE;
   v_caregiver_name Caregiver.caregiver_name%TYPE;
   v_package_name DaycarePackages.package_name%TYPE;
BEGIN
   OPEN c_caregivers_packages;
   LOOP
      FETCH c_caregivers_packages INTO v_caregiver_id, v_caregiver_name, v_package_name;
```

```
        EXIT WHEN c_caregivers_packages%NOTFOUND;
        DBMS_OUTPUT.PUT_LINE('Caregiver: ' || v_caregiver_name || ', Package: ' ||
v_package_name);
    END LOOP;
    CLOSE c_caregivers_packages;
END;
```

2. Increase Daycare Package Cost by 10% for Packages Under $200:

```
SET SERVEROUTPUT ON;
BEGIN
    UPDATE DaycarePackages
    SET package_cost = package_cost * 1.1
    WHERE package_cost < 200;

    DBMS_OUTPUT.PUT_LINE('Daycare package costs updated.');
END;
/
```

3. Trigger to Ensure that Caregivers Do Not Exceed Their Maximum Capacity :

```
CREATE OR REPLACE TRIGGER trg_check_caregiver_capacity
BEFORE INSERT ON Child
FOR EACH ROW
DECLARE
```

```
  v_current_count NUMBER;
  v_max_capacity NUMBER := 5; -- Assuming each caregiver can handle up to 5 children
BEGIN
  SELECT COUNT(*) INTO v_current_count
  FROM Child
  WHERE caregiver_id = :NEW.caregiver_id;

  IF v_current_count >= v_max_capacity THEN
    RAISE_APPLICATION_ERROR(-20001, 'This caregiver has reached their maximum capacity.');
  END IF;
END;
/
```

Users of Daycare Database Management:

The users or targeted customers of a daycare management system database typically encompass a range of stakeholders involved in child care and education. Here are the key user groups:

1. **Daycare Administrators:** Administrators at daycare centers use these systems to oversee daily operations, staff schedules, and child enrollment. They rely on the system to maintain orderly records and ensure compliance with safety and educational standards.
2. **Caregivers and Teachers:** Caregivers and teachers within the daycare use the system to access daily schedules, child information, and communication logs with parents. This helps them provide personalized care and educational activities based on the specific needs and development stages of each child.

3. **Parents:** Parents are direct users of the system for registering their children, viewing daily schedules, receiving updates about their child's activities, and communicating with caregivers. The system provides a transparent view of the child's day-to-day care and development.

4. **Healthcare Professionals:** Pediatricians or healthcare providers may access certain parts of the system, such as health records or daily activity reports, to monitor the well-being of children and provide necessary interventions.

5. **Regulatory Bodies:** Governmental bodies and regulatory agencies might use the system to ensure that daycare centers comply with childcare standards and regulations. They can audit records, monitor compliance, and ensure the safety and quality of care provided.

These users rely on the daycare management system to ensure efficient, safe, and nurturing environments that support the developmental needs of children while also facilitating smooth operational workflows for daycare providers.

Conclusion:

In conclusion, the Daycare Management System (DMS) stands as a groundbreaking solution designed to streamline the complexities of managing daycare operations. This system is meticulously engineered to handle detailed records of caregivers, parents, and children, ensuring that daycare centers can provide personalized and efficient care. By organizing crucial data such as daily schedules, child development activities, and caregiver assignments, the DMS allows for enhanced operational efficiency and improved communication between daycare staff and parents.

Moreover, the DMS's ability to integrate complex SQL queries and database procedures empowers administrators to perform detailed analysis and maintain high standards of safety and quality in child care. This adaptability in managing diverse data helps in complying with educational standards and regulatory requirements, thus maintaining the daycare center's integrity and trust among parents.