
Software Requirements Specification

for

Game Zone

Version 1.0 approved

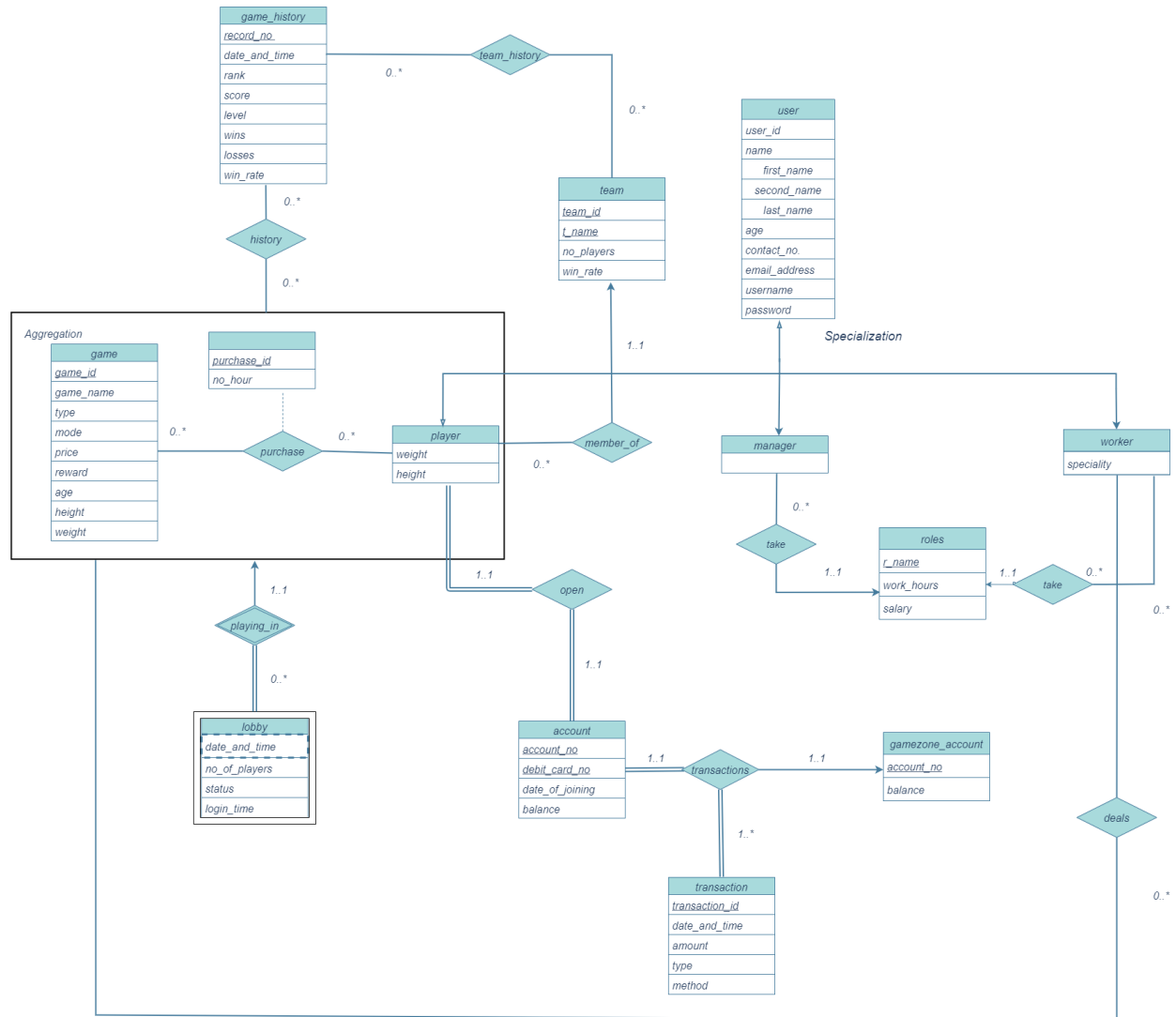
Prepared by:

Thakkar Amit (201901038)
Makwana Jigar (201901428)
Prajapati Parth (201901429)
Gohil Arpit (201901471)

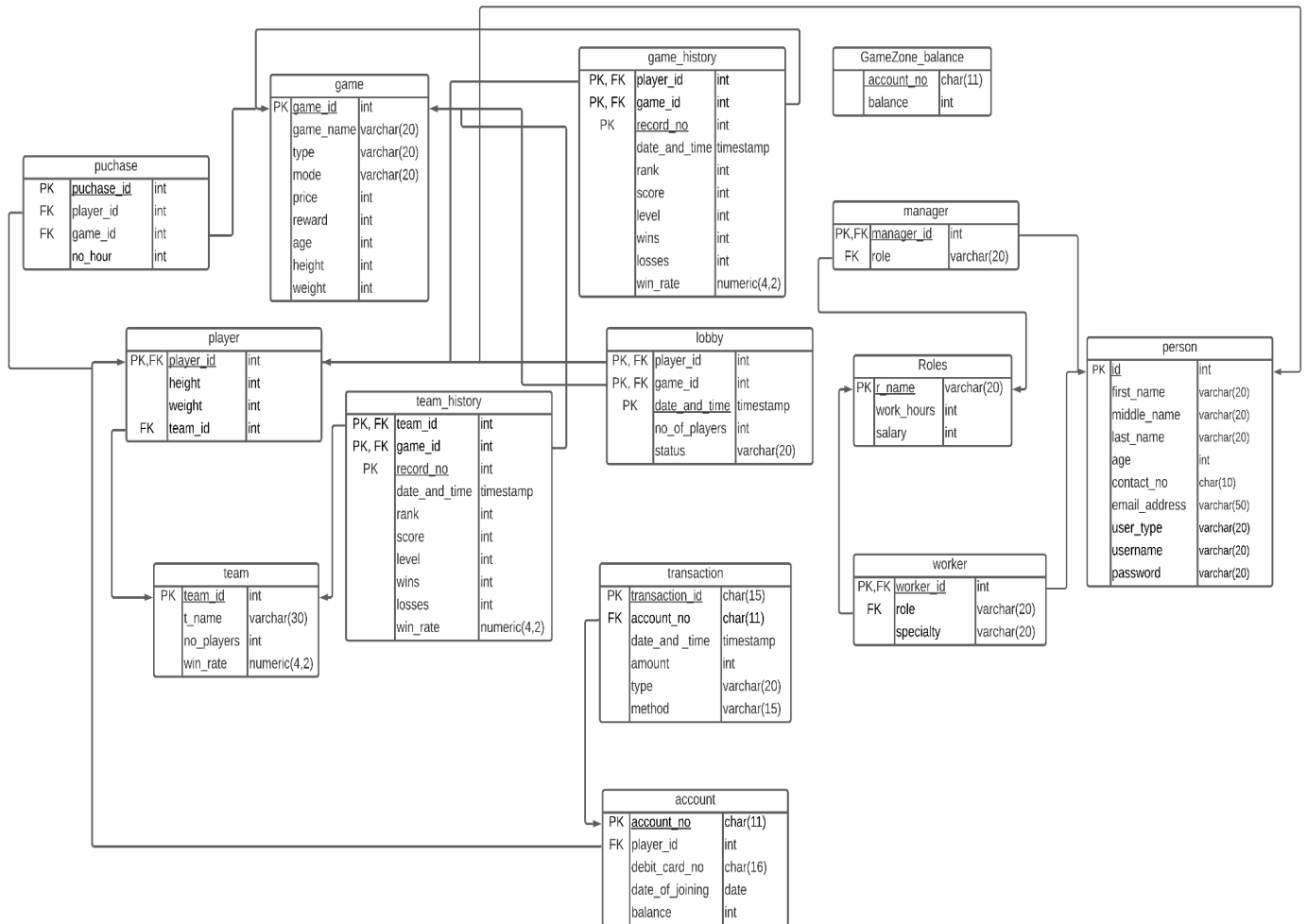
DAIICT

30/10/2021

ER Diagram:



Relational Model:



SCHEMA:

- ❖ game(**game_id**, game_name, type, mode, price, reward, age, height, weight)
- ❖ game_history(**record_no**, game_id, player_id, date_and_time, rank, score, level, wins, losses, win_rate)
- ❖ purchase(**purchase_id**, game_id, player_id, no_hour)
- ❖ player(**player_id**, height, weight, team_id)
- ❖ team_history(**record_no**, team_id, game_id, rank, date_and_time, score, level, wins, losses, win_rate)
- ❖ lobby(**date_and_time**, player_id, game_id, no_of_players, status, login_time)
- ❖ manager(**manager_id**, role)
- ❖ roles(**r_name**, work1_hours, salary)
- ❖ transactions(**transaction_id**, account, date_and_time, amount, type, method)
- ❖ team(**team_id**, t_name, no_players, win_rate)
- ❖ account(**account_no**, player_id, debit_card_no, date_of_joining, balance)
- ❖ worker(**worker_id**, role, speciality)
- ❖ person(**id**, first_name, middle_name, last_name, contact_no, age, email_address, user_type, username, password)
- ❖ GameZone_balance(account_no, balance)

DDL Scripts:

1. Person

```
CREATE TABLE IF NOT EXISTS person (  
    id int,  
    first_name varchar(20) COLLATE pg_catalog."default"  
not null,  
    middle_name varchar(20) COLLATE pg_catalog."default"  
not null,  
    last_name varchar(20) COLLATE pg_catalog."default"  
not null,  
    age int not null,  
    contact_no char(10),  
    email_address varchar(20) COLLATE  
pg_catalog."default" not NULL UNIQUE,  
    user_type varchar(20) COLLATE pg_catalog."default"  
not null,  
    username varchar(20) COLLATE pg_catalog."default"  
not null UNIQUE,  
    password varchar(20) COLLATE pg_catalog."default"  
NOT NULL,  
    PRIMARY KEY (id),  
    CHECK(user_type in ('manager','player','worker'))  
);
```

2. team

```
CREATE TABLE IF NOT EXISTS team (  
    team_id int,  
    t_name varchar(30) not null unique,  
    no_players int not null CHECK(no_players > 1),  
    win_rate numeric(4,2) default 0,  
    PRIMARY KEY (team_id)  
);
```

3. Roles

```
CREATE TABLE IF NOT EXISTS Roles (  
    r_name varchar(20) COLLATE pg_catalog."default" NOT  
    NULL UNIQUE,  
    workhours int not null check(workhours > 0),  
    salary int not null check(salary > 0),  
    PRIMARY KEY (r_name)  
);
```

4. player

```
CREATE TABLE IF NOT EXISTS player (  
    player_id int,  
    height int not NULL,  
    weight int not NULL,  
    team_id int,
```

```
PRIMARY KEY (player_id),  
FOREIGN KEY (player_id) REFERENCES person(id) MATCH  
SIMPLE ON UPDATE CASCADE ON DELETE CASCADE,  
FOREIGN KEY (team_id) REFERENCES team(team_id) MATCH  
SIMPLE ON UPDATE CASCADE ON DELETE CASCADE  
);
```

5. manager

```
CREATE TABLE IF NOT EXISTS manager (  
    manager_id int,  
    role varchar(20) COLLATE pg_catalog."default" NOT  
NULL,  
    PRIMARY KEY (manager_id),  
    FOREIGN KEY (role) REFERENCES roles(r_name) MATCH  
SIMPLE ON UPDATE CASCADE ON DELETE CASCADE,  
    FOREIGN KEY (manager_id) REFERENCES person(id) MATCH  
SIMPLE ON UPDATE CASCADE ON DELETE CASCADE  
);
```

6. worker

```
CREATE TABLE IF NOT EXISTS worker (  
    worker_id int,  
    role varchar(20) COLLATE pg_catalog."default" not  
NULL,  
    specialty varchar(20) COLLATE pg_catalog."default"  
not NULL,
```

```
PRIMARY KEY (worker_id),  
FOREIGN KEY (role) REFERENCES roles(r_name) MATCH  
SIMPLE ON UPDATE CASCADE ON DELETE CASCADE,  
FOREIGN KEY (worker_id) REFERENCES person(id) MATCH  
SIMPLE ON UPDATE CASCADE ON DELETE CASCADE  
);
```

7. account

```
CREATE TABLE IF NOT EXISTS account (  
account_no char(11),  
player_id int not NULL UNIQUE,  
debit_card_no char(16) not NULL ,  
date_of_joining date not NULL,  
balance int default 1000,  
PRIMARY KEY (account_no),  
FOREIGN KEY (player_id) REFERENCES player(player_id)  
MATCH SIMPLE ON UPDATE CASCADE ON DELETE CASCADE  
);
```

8. transactions

```
CREATE TABLE IF NOT EXISTS transactions (  
transaction_id char(15),  
account_no char(11) not NULL UNIQUE,  
date_and_time timestamp not NULL,  
amount int not NULL,
```



```

    type varchar(20) COLLATE pg_catalog."default" not
NULL,
    method varchar(15),
    PRIMARY KEY (transaction_id),
    FOREIGN KEY (account_no) REFERENCES
account(account_no) MATCH SIMPLE ON UPDATE RESTRICT ON
DELETE RESTRICT
);

```

9. game

```

CREATE TABLE IF NOT EXISTS game (
    game_id int,
    game_name varchar(20) COLLATE pg_catalog."default"
not null unique,
    type varchar(20) COLLATE pg_catalog."default" ,
    mode varchar(20) COLLATE pg_catalog."default" not
null,
    price int not null CHECK(price > 0),
    reward int default 0,
    age int default 5,
    height int,
    weight int,
    PRIMARY KEY (game_id),
    CHECK(mode in ('online','offline'))
);

```

10. purchase

```

CREATE TABLE IF NOT EXISTS purchase (
    purchase_id int,
    player_id int,
    game_id int ,
    no_hours int not null CHECK(no_hours > 0),
    PRIMARY KEY (purchase_id),
    FOREIGN KEY (player_id) REFERENCES
player(player_id) MATCH SIMPLE ON UPDATE CASCADE ON
DELETE CASCADE,
    FOREIGN KEY (game_id) REFERENCES game(game_id)
MATCH SIMPLE ON UPDATE CASCADE ON DELETE CASCADE
);

```

11. team_history

```

CREATE TABLE IF NOT EXISTS team_history (
    team_id int,
    game_id int,
    record_no int,
    date_and_time timestamp not NULL,
    rank int NOT NULL CHECK(rank > 0),
    score int not NULL CHECK(score >=0),
    level int default 0 CHECK(level > 0),
    wins int NOT NULL CHECK(wins > 0),
    losses int NOT NULL CHECK(losses > 0),
    win_rate numeric(4,2) NOT NULL,
    PRIMARY KEY (team_id, game_id, record_no),

```

```

    FOREIGN KEY (game_id) REFERENCES game(game_id)
MATCH SIMPLE ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (team_id) REFERENCES team(team_id)
MATCH SIMPLE ON UPDATE CASCADE ON DELETE CASCADE
);

```

12. game_history

```

CREATE TABLE IF NOT EXISTS game_history (
    player_id int,
    game_id int,
    record_no int,
    date_and_time timestamp not NULL,
    rank int NOT NULL CHECK(rank > 0),
    score int not NULL CHECK(score >=0),
    level int CHECK(level > 0),
    wins int NOT NULL CHECK(wins > 0),
    losses int NOT NULL CHECK(losses > 0),
    win_rate numeric(4,2) NOT NULL,
    PRIMARY KEY (player_id, game_id, record_no),
    FOREIGN KEY (game_id) REFERENCES game(game_id),
    FOREIGN KEY (player_id) REFERENCES
player(player_id)
);

```

13. lobby

```

CREATE TABLE IF NOT EXISTS lobby (
    player_id int,
    game_id int,
    date_and_time timestamp,
    no_of_players int not null CHECK(no_of_players > 0),
    status varchar(20) COLLATE pg_catalog."default" not
NULL,
    login_time time not null,
    PRIMARY KEY (player_id, game_id, date_and_time),
    FOREIGN KEY (player_id) REFERENCES
player(player_id) MATCH SIMPLE ON UPDATE CASCADE ON
DELETE CASCADE,
    FOREIGN KEY (game_id) REFERENCES game(game_id)
MATCH SIMPLE ON UPDATE CASCADE ON DELETE CASCADE,
    CHECK (status in ('playing','waiting'))
);

```

14. GameZone_balance

```

CREATE TABLE IF NOT EXISTS GameZone_balance (
    account_no char(11) NOT NULL,
    balance int default 50000
);

```

-----X--X-----