Ahmed, Ayesha
Warman, Emily
Caron, Andrew
CS425 Final Project

CS425 Design Document

Group Name: TCBT (The Country's Best Theaters)

Feature ownership:

Andrew Caron: made the GUI interface for users, checked table structure, connected backend with frontend, wrote queries, wrote insert-update-delete queries, wrote the Test Document.
Ayesha Ahmed: (EDITOR) restructured and created tables, inserted test data, wrote backend Java code for GUI information parsing, integrated some backend functions with the frontend GUI, wrote the Design Document, Demo Slides and README.
Emily Warman: wrote backend Java code for GUI information parsing, integrated backend functions with the frontend GUI, made github project, wrote queries, checked table structure.

Table Descriptions:

This is a simple breakdown description of the database tables we used to make our project. The description details the table name, attributes, keys, functional dependencies, and normal forms of each table. The tables used for homework 3 part 2 were not changed.

Locations(theater_id, address, city, state, zip, theater_name)
primary key (theater_id)
FD: theater_id -> address, city, state, zip, theater_name.
BCNF

TheaterInfo(room_id, theater_id, room_num, capacity)
unique (theater_id, room_num),
foreign key (theater_id) REFERENCES Locations(theater_id),
primary key (room_id)
FD: room_id -> theater_id, room_num, capacity
BCNF

Movies(movie_id, title, DirectorName, Description)
unique (title, DirectorName),
Primary Key (movie_id)

FD: movie_id -> title, DirectorName, Description
BCNF

Stars(movie_id, StarName)
FOREIGN KEY (movie_id) REFERENCES Movies(movie_id),
PRIMARY KEY(StarName, movie_id)
FD: (StarName, movie_id)
BCNF

Genre(movie_id, GenreType)
FOREIGN KEY (movie_id) REFERENCES Movies(movie_id),
primary key (GenreType, movie_id)
FD: (GenreType, movie_id)
BCNF

Schedule(showing_id, movie_id, room_id, show_date, ticket_price, tickets_sold)
foreign key (movie_id) references Movies(movie_id),
FOREIGN KEY (room_id) REFERENCES TheaterInfo(room_id),
PRIMARY KEY (showing_id)
FD: showing_id -> movie_id, room_id, show_date, ticket_price, tickets_sold
BCNF

CC(ccn, ccn_code, cc_name, card_type, exp_date, street1, street2, city, state, zip)
primary key (ccn)
FD: ccn -> ccn_code, cc_name, card_type, exp_date, street1, street2, city, state, zip
BCNF

CreditCardCompany(ccn, cc_balanace)
foreign key (ccn) references CC(ccn),
primary key (ccn)
FD: ccn -> cc_balance

TheaterUsers(username, password, name, ccn, phone, email)
foreign key (ccn) references CC(ccn),
primary key (ccn)
FD: ccn -> username, password, name, phone, email
BCNF

Tickets(ccn, showing_id, ticket_no, date_purchased)
foreign key (ccn) references TheaterUsers(ccn),
foreign key (showing_id) references Schedule(showing_id),
primary key (ccn, showing_id, date_purchased)

FD: (ccn, showing_id, date_purchased) -> ticket_no
BCNF

Points(username, current_points, total_points)
foreign key (username) references TheaterUsers(username),
primary key (username)
FD: username -> current_points, total_points

PointLevel(level_name, level_boundary)
primary key (level_name)
FD: level_name -> level_boundary
BCNF

Rewards(theater_id, movie_points, review_points, level_name, offers, deals)
foreign key (level_name) references PointLevel(level_name),
foreign key (theater_id) REFERENCES Locations(theater_id),
primary key (theater_id, level_name)
FD: (theater_id, level_name) -> movie_points, review_points, offers, deals
BCNF

TheaterThreads(id, theater_id, username, text)
foreign key (username) references TheaterUsers(username),
foreign key (theater_id) REFERENCES Locations(theater_id),
primary key (id)
FD: id -> theater_id, username, text
BCNF

TheaterComments(thread_id, comment_number, username, text)
foreign key (username) references TheaterUsers(username),
foreign key (thread_id) references TheaterThreads(id),
primary key (thread_id, comment_number)
FD: (thread_id, comment_number) -> username, text
BCNF

MovieThreads(id, username, movie_id, movie, star_name, director, text)
foreign key (username) references TheaterUsers(username),
foreign key (movie, director) references Movies(title, DirectorName),
foreign key (star_name,movie_id) references Stars(StarName,movie_id),
primary key (id)
FD: id -> username, movie_id, movie, star_name, director, text
BCNF

MovieComments(thread_id, comment_number, username, text)
foreign key (username) references TheaterUsers(username),
foreign key (thread_id) references MovieThreads(id),
primary key (thread_id, comment_number)
FD: (thread_id, comment_number) -> username, text
BCNF

JobTypes(job_type, description)
primary key (job_type)
FD: job_type -> description
BCNF

Management(manager_id, theater_id, man_type, username, sched_password, man_name, address, phone, ssn)
foreign key (theater_id) REFERENCES Locations(theater_id),
foreign key (man_type) REFERENCES JobTypes(job_type),
primary key (manager_id)
FD: manager_id -> theater_id, man_type, username, sched_password, man_name, address, phone, ssn
BCNF

Employees(emp_id, emp_name, address, phone, ssn, hiredby_id)
foreign key (hiredby_id) references Management(manager_id),
primary key (emp_id)
FD: emp_id -> emp_name, address, phone, ssn, hiredby_id
BCNF

JobTraining(emp_id, janitor, salesRep, ticketMaster)
foreign key (emp_id) references Employees(emp_id),
primary key (emp_id)
FD: emp_id -> janitor, salesRep, ticketMaste
BCNF

EmpSchedule(emp_id, job_date, theater_id, job_type)
unique(emp_id, job_date),
foreign key (emp_id) references Employees(emp_id),
foreign key (job_type) references JobTypes(job_type),
foreign key (theater_id) REFERENCES Locations(theater_id),
primary key (job_date, emp_id)
FD: (job_date, emp_id) -> theater_id, job_type
BCNF

Ahmed, Ayesha
Warman, Emily
Caron, Andrew
CS425 Final Project

Entity Relation Diagram:

This diagram showcases the above listed relationships between our tables in a visual format.