Databases Project

**Topic**

*Train information: train schedules, stations, times, tickets.*

**Topic Analysis**

Client: SKM (Szybka Kolej Miejska w trójmieście, division of the PKP company) president

Purpose: Providing easy access to current SKM trains schedule, arrival times of actual runs that could be delayed and passengers tickets data, improving passenger experience and decreasing difficulty of operations needed to be done by all workers related to SKM

Current Problems: Passengers, Train workers and Dispatchers do not have a stable and efficient information on current schedules and possible delays of SKM trains making communication and reliable planning difficult. Ticket sellers do not need to waste resources by printing tickets, when they can just update the tickets database and ticket inspectors can have trouble with checking some tickets purchased by passengers (damaged tickets, online tickets on damaged or out of battery devices)

System Users and their examples of Scenarios of Use / Queries:

1. Passengers

Scenario: Searching the current train schedules, Purchasing tickets, Viewing bought tickets

Query: Show the price of ticket from A to B with specific discount, Show soonest departures from one station, Show how long it takes to commute from A to B, What are the train stops on the route

1. Ticket Sellers

Scenario: Adding purchased tickets to the ticket database, Evaluating ticket prices based on length of the passenger route and their possible discounts

Query: Check ticket price on route section, Upload a new valid ticket to ticket database

1. Train Workers

Scenario: Uploading the exact time of stops at each station, Evaluating difference from planned times of arrival (How late/early the train came to the station)

Query: Upload time of arrival, Check train delays, Check average delay on specific route

1. Dispatchers

Scenario: Checking the possible variances in planned time of arrival of the trains, Checking the platform they should direct the train to

Query: Check train delays, Check platform on which the soonest train should stop at, Display all currently active trains on specific route

1. Ticket Inspectors

Scenario: Checking the validity of passenger tickets, Uploading tickets bought in train

Query: Check tickets of specific passenger, Upload a new valid ticket to ticket database, Check if current station is included in the passenger ticket

Assumptions:

* Database will replace previous systems, collecting everything in one place
* Database will be able to manage changes in train schedules
* Database will be accessible on both desktop and mobile devices
* Database will contain data on every relevant ticket and run of SKM train

Limitations:

* Database will contain data only for trains belonging to SKM division
* Access to ticket data limited for not authorized entities (Passengers can only view their own tickets)
* Ticket data, Time of actual stop times of every SKM train and data of specific train run is held no longer than 3 months
* Passenger data removed after a year of inactivity

**Description of ERD Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 1: SKM\_Routes** | | | |
| **Description** | | | |
| Number of Records: around 20  Represents all main and temporary lines of SKM. Route creates a connection between the starting and ending station of the line. Entity set collects most important information on the routes, like time of completion, starting / ending point etc. Entries deleted after one year of line being discontinued or instantly if there is a change to the line number they represent. Entities entered during creation of new line or modification of the currently active line | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| line\_number | Alphanumeric | Yes | Unique Alphanumeric text consisting of uppercase letter “S” and integer number up to 3 digits |
| starting\_station | Positive Integer | No | Integer representing ID of the station (up to 100) |
| ending\_station | Positive Integer | No | Integer representing ID of the station (up to 100) |
| travel\_time | Positive number with two decimal places | No | Represents number of minutes (30 seconds is 0.5 and so on, between 0.00 and 600.00) |
| num\_of\_stops | Positive Integer | No | Quantity of stops needed to complete the route (between 1 and 100) |
| distance\_km | Positive number with two decimal places | No | Number of kilometers needed to complete the route (between 0.00 and 1000.00) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 2: Planned Schedules** | | | |
| **Description** | | | |
| Number of Records: up to around 2000  Represents all weekly planned departures for each route and departures on unique days that result in schedule change (for example holidays, maintenance). Entities deleted after one year of being discontinued. Entities entered when uploading new time schedule (could be related to maintenance, holidays or if just changing the normal weekly schedule of departures) | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| line\_number | Alphanumeric | No | Alphanumeric text consisting of uppercase letter “S” and integer number up to 3 digits |
| departure\_time | Time | No | Time shown in HH:MM:SS format |
| arrival\_time | Time | No | Time shown in HH:MM:SS format |
| day\_of\_the\_week | String | No | String representing specific day of the week where this schedule is planned for departure, being = “Monday” or “Tuesday” or … |
| day\_type | String | No | “Weekday” or “Weekend” or “Holiday” or “Maintenance” or specification of which exact holiday or type of maintenance it is (up to 100 characters) |
| valid\_from | Date | No | Date in YYYY-MM-DD format |
| valid\_to | Date | No | Date in YYYY-MM-DD format |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 3: Runs** | | | |
| **Description** | | | |
| Number of Records: around 24 000  Collecting data of all actual SKM train runs for the latest quarter. Entries deleted after being older than 3 months. Entities entered when uploading new run of the SKM trains | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| schedule\_id | Positive Integer | No | Links to planned schedules - Business key going up to around 2000 |
| run\_date | Date | No | Date of departure in YYYY-MM-DD format |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 4: Stations** | | | |
| **Description** | | | |
| Number of Records: up to around 100  Represents every station that could be a stop for the SKM trains. Entries deleted when a particular station is no longer active. Entities entered when the new station accepts stops of SKM trains. | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| station\_name | String | No | String containing up to 50 characters (only letters in polish alphabet and spaces allowed, every word starts with uppercase character) |
| num\_of\_platforms | Positive Integer | No | Integer number in range 1 to 15 |
| station\_address | String | No | Address of the street the Station is located at (up to 100 characters) |
| phone\_number | String | No | Formatted as (+DD-DDD-DDD-DDD) where D stands for digit |
| city | String | No | String containing up to 50 characters (only letters in polish alphabet and spaces allowed) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 5: Route Stops** | | | |
| **Description** | | | |
| Number of Records: up to around 600  This entity set represents each stop in correct order on every SKM route. It is a template showing how a particular route should ideally be completed. Entries removed when the route stops on their line change or one year after the route line has been discontinued. Entities entered when appending / modifying the route stops for a specific line | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| line\_number | Alphanumeric | Yes | Alphanumeric text consisting of uppercase letter “S” and integer number up to 3 digits |
| stop\_number | Positive Integer | Yes | Represents the sequence of stops (range from 1 to 100) |
| station\_id | Positive Integer | No | Representing station id (range from 1 to around 100) |
| current\_km | Positive number with two decimal places | No | Number of kilometers from start of the route (between 0.00 and 1000.00) |
| current\_travel\_time | Positive number with two decimal places | No | Represents number of minutes passed from departure (30 seconds is 0.5 and so on, between 0.00 and 600.00) |
| platform\_num | Positive Integer | No | Integer number in range 1 to 15, representing the default platform number the train should stop at |
| stop\_duration | Positive number with two decimal places | No | Number of minutes planned for waiting at the station (30 seconds is 0.5 and so on, between 0.00 and 60.00) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 6: Tickets** | | | |
| **Description** | | | |
| Number of Records: around 10 000 000  Contains data on all tickets purchased for SKM train rides the last 3 months. After 3 months the entity is deleted. Entities entered when a ticket is purchased | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| passenger\_id | Positive Integer | No | Representing the passenger owning the ticket |
| discount\_id | Positive Integer | No | Representing the type of discount |
| run\_id | Positive Integer | No | Representing exact SKM train ride that the ticket is purchased for |
| date\_of\_purchase | Date | No | Date in YYYY-MM-DD format |
| starting\_station | Positive Integer | No | Integer representing ID of the station (up to around 100) |
| ending\_station | Positive Integer | No | Integer representing ID of the station (up to around 100) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 7: Passengers** | | | |
| **Description** | | | |
| Number of Records: around 1 000 000  Collection of all ticket owners and their essential personal data. Record of passenger deleted after one year of not owning a ticket. Entities entered when passenger not already included in this entity set purchases a ticket | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| passenger\_name | String | No | First name of passenger (up to 50 characters, allowed only letters and spaces, all words start with uppercase character) |
| passenger\_surname | String | No | Last name of passenger (up to 55 characters, allowed only letters and spaces, all words start with uppercase character) |
| date\_of\_birth | Date | No | Date in YYYY-MM-DD format |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 8: Discounts** | | | |
| **Description** | | | |
| Number of Records: around 30  Stores all types of discounts available for SKM passengers that are not expired for longer than one year. Entities entered when a new type of discount is accepted for SKM train rides. Entities deleted a year after finish date | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| name\_of\_type | String | No | Name describing the discount (up to 50 characters |
| discount\_percentage | Positive number with two decimal places | No | Percentage of price removed from base price of the ticket (range from 0.00 to 1.00) |
| requirements | String | No | Exact description providing information on requirements needed to be fulfilled to use the discount (up to 700 characters) |
| finish\_date | Date | No | Date in YYYY-MM-DD format or NULL value if discount is said to be non-expiring |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 9: Base Prices** | | | |
| **Description** | | | |
| Number of Records: around 21  Represents normal ticket (without discounts) price for each of the possible ride length ranges. Price of the normal ticket for the SKM ride depends only on the length of the ride from the starting station to the ending station. Records modified / deleted only based on fare changes. Entities entered when there is a change in base prices or a new distance range possible | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| range\_id | Positive Integer | Yes | Positive integer business key, starting from one, increasing by one |
| price | Positive number with two decimal places | No | Range from 0.00 to 400.00 |
| lower\_bound | Positive Integer | No | Represents the minimum amount of kilometers for the ride to be belonging to the range (between 0 and 1000) |
| upper\_bound | Positive Integer | No | Represents the maximum amount of kilometers for the ride to be belonging to the range (between 0 and 1000) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Entity Set 10: Stop Times** | | | |
| **Description** | | | |
| Number of Records: around 720 000  This entity set stores data on each particular stop on every actual run made by SKM trains in the last 3 months. After 3 months from the date of the run all stop times from this run are removed. Entities entered when a train on the run stops at a station | | | |
| **Attributes** | | | |
| **Name** | **Type** | **Primary Key** | **Description** |
| run\_id | Positive Integer | Yes | ID of the run connected to the stop (range from 1 up to around 24 000) |
| stop\_number | Positive Integer | Yes | Represents the sequence of stops ( range from 1 to 100) |
| station\_id | Positive Integer | No | Representing a station where this stop is located at ( range from 1 to around 100) |
| stop\_duration | Positive number with two decimal places | No | Number of minutes the train was waiting at the station (30 seconds is 0.5 and so on, between 0.00 and 1440.00) |
| time\_of\_stop | Date and Time | No | Date and Time when the train stopped at the station (in format  YYYY-MM-DD HH:MM:SS ) |
| platform\_num | Positive Integer | No | Integer number in range 1 to 15, representing the platform the train stopped at |

**Description of the relationship**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **First Entity** | **Type** | **Second Entity** | **Description** |
| Have A | Discounts | 0..1 : 0..n | Tickets | Represents whether the ticket contains a discount. Tickets can but do not need to be discounted. One discount can be applied to many tickets, but there may be a situation when nobody used a specific discount. Needed for evaluating ticket price |
| Owns A | Passengers | 1 : 0..n | Tickets | Links a person to the ticket. Every ticket needs to be assigned to exactly one specific passenger. A passenger can purchase many tickets, but also can be assigned to none (for example after 3 months tickets are removed from the database, but inactive passengers after a year). Needed for things such as checking tickets during train ride |
| Stops At | SKM\_Routes | 1 : 0..n | Route Stops | Allows for showing every single stop on a specific SKM line route. Every route stop must be assigned to a route that it stops at. The route usually has several stops on the way, but in case of creating a new route, at first it may not have any selected stops. Needed for checking the path of a route |
| Goes On | Planned Schedules | 0..n : 1 | SKM\_Routes | Allows for linking a specific schedule of departures of SKM trains to the routes all scheduled rides will go on. Every scheduled departure needs to go on a specific route, but the route does not need to have any departures planned on it. Needed for checking what path every scheduled train will go on |
| Stops At | Route Stops | 0..n : 1 | Stations | Allows for checking station information for each stop on a specific route. Every route stop needs to be on some specific station, but not every station needs to have a current route stopping there. Needed for checking where actually a train route stops at |
| Have A | Base Prices | 1 : 0..n | Tickets | Every ticket is connected to some based price evaluated by checking the length of the ride this ticket is allowed on. There needs to be some price connected to every ticket, but there may be no tickets purchased for a specific length range, which results in optionality. Needed for calculating ticket price |
| Starts At | Tickets | 0..n : 1 | Stations | Allows for showing station information on where the ticket starts at. Every ticket must have a starting and ending station, but not every station needs to have a recent ticket that starts the journey there. Needed to show the name of the station that the tickets start at |
| Ends At | Tickets | 0..n : 1 | Stations | Allows for showing station information on where the ticket ends at. Every ticket must have a starting and ending station, but not every station needs to have a recent ticket that ends the journey there. Needed to show the name of the station that the tickets ends at |
| Starts At | SKM\_Routes | 0..n : 1 | Stations | Allows for showing station information on where the route starts at. Every route must have a starting and ending station, but not every station needs to have a route that starts the journey there. Needed to show the name of the station that the route starts at |
| Ends At | SKM\_Routes | 0..n : 1 | Stations | Allows for showing station information on where the route ends at. Every route must have a starting and ending station, but not every station needs to have a route that ends the journey there. Needed to show the name of the station that the route ends at |
| Goes On | Runs | 0..n : 1 | Planned Schedule | Allows for linking a specific SKM run to the schedule of departures of SKM trains. Every run needs to go on a specific schedule, but a planned schedule does not need to have any recent runs on it. Needed for checking whether the train is late / early or if something went wrong during the run |
| Stopped At | Runs | 1 : 0..n | Stop Times | Links every actual stop to the run it happened at. Every stop needs to be connected to a specific run, but for example when a run just got created it will not have any stop times added yet. Needed for writing down the actual times the train stopped at a station, which allows for operations like checking the time difference from the schedule |
| For A | Tickets | 0..n : 1 | Runs | Links tickets to the train ride they are bought for. Every ticket needs to be attached to a specific SKM train run, but some runs may have no passengers on them. Needed for example for checking ticket validity |
| Stops At | Stop Times | 0..n : 1 | Stations | Allows for showing station information on where the run stopped at. Every stop must be at a specific station, but not every station needs to have a SKM train stop there. Needed to show the name of the station that the SKM train stopped at |

**Relational Database Schema**

SKM\_Routes (line\_number, starting\_station ref stations, ending\_station ref stations, travel\_time, num\_of\_stops, distance\_km)

Planned\_Schedules (id, line\_number ref SKM\_Routes, departure\_time, arrival\_time, day\_of\_the\_week, day\_type, valid\_from, valid\_to)

Runs (id, schedule\_id ref planned schedules, run\_date)

Stations (id, station\_name, num\_of\_platforms, station\_address, phone\_number, city)

Route\_Stops (line\_number ref SKM\_Routes, stop\_number, station\_id ref stations, current\_km, current\_travel\_time, platform\_num, stop\_duration)

Tickets (id, passenger\_id ref passengers, discount\_id ref discounts, run\_id ref runs, date\_of\_purchase, starting\_station ref stations, ending\_station ref stations, normal\_price ref base prices)

Passengers (id, passenger\_name, passenger\_surname, date\_of\_birth)

Discounts (id, name\_of\_type, discount\_percentage, requirements, finish\_date)

Base\_Prices (range\_id, price, lower\_bound, upper\_bound)

Stop\_Times (run\_id ref runs, stop\_number, station\_id ref stations, stop\_duration, time\_of\_stop, platform\_num)

**ERD Diagram**

All attribute names should be written using lowercase letters instead of uppercase

