

## Public Bus Transport System (PBTs)

### 1. Brief Description of the Project

PBTs (Public Bus Transport System) is Singapore's system for providing efficient and safe public bus transport. One key mission for PBTs is to use a data-driven approach for analyzing travel patterns and improving transport services. The objective of this course project is for student teams to develop a PBTs database system that can support the key features specified in this document.

### 2. Key Features of the Project.

To simplify the course project, the following are the only functions that your PBTs database system must support. If any of the functions are different from the one offered by the actual PBTs system, please adhere to the functions specified in this document. **Please ignore other functions that actual PBTs may offer.**

The attributes that your system must keep track of are specified in this document: your system should not keep track of any attribute that is not mentioned in this document unless it is necessary to support the listed required functions in this document. Derived attributes, if any, should be included in your proposed ER diagram.

#### 2.1. Function 1: Bus service information

A bus stop provides bus services to passengers traveling around Singapore. There are many bus stops that are of interest to PBTs and will be recorded. A bus stop is uniquely identified by a stop ID. PBTs stores the geospatial location of the bus stop as address and description of the location. Each bus stop records all bus services available at the bus stop.

A bus service is uniquely identified by a service ID. Note that some bus services have two directions, e.g., Bishan to City Hall, and the opposite City Hall to Bishan. Each direction of a bus service is considered as an individual service that has its own service ID. PBTs also tracks the order of the bus stops served by a bus service. To simplify the project, we assume a bus service never passes through the same bus stop twice. Furthermore, we need to know the total

number of stops for every bus service. Each bus service is operated by one or more bus companies. We need to store the company (or companies) that operate the bus service. Note that each company has a unique name. For a bus service, it is either operating on a normal route or an express route. For a normal route, PBTS stores the bus frequency of the service on weekdays and its frequency on weekends. For an express route, PBTS stores if the service is available on weekends and also whether it operates on public holidays. There could be some services which are available on weekends but not on public holidays.

For each bus service, PBTS should support queries on all the connecting bus services, i.e., bus services that share at least one common bus stop with it. For providing convenient travel support to passengers, PBTS should also support queries on any pair of bus stops that are connected by direct bus services, i.e., no bus transition is needed to travel from one to the other.

## **2.2. Function 2: Service operation information**

A bus is uniquely identified by its plate number. PBTS stores the fuel type of the bus (gas or electric) and the capacity of the bus. One bus can serve any number of bus services and a bus operates a bus service as a bus trip. There can be many bus trips for a bus service but the date and start time of the trip will be different for each bus trip. PBTS stores the date of the bus trip, the starting time, and the completion time of the bus trip (we assume that the starting time and the completion time are on the same date; ignore cases where the trip starts before midnight and ends after midnight). PBTS also tracks the driver for each bus trip. The driver information includes a unique DID, a unique NRIC number, name, date of birth (DoB), and gender. Note that a driver may drive different buses and a bus can switch drivers even within a day.

## **2.3. Function 3: CityLink card information**

Passengers use CityLink cards to pay for the bus services. A CityLink card has a unique card ID and PBTS also stores the remaining balance of the card. There are three types of CityLink cards: Adult card, Senior card, and Student card. For each type of card, PBTS stores the discount (if any), the minimum top-up amount, and the eligibility description of the card type. The system needs to know the type of CityLink card to charge the correct amount of fees when

the passenger uses the transport services. A CityLink card has an expiry date, which has to be stored. Once the card expires, a new card of the same type can be purchased to replace the expired card and enjoy the same features (e.g., discount); the new card has its own new card ID, and the database needs to record which card it has replaced.

## **2.4. Function 4: Passenger travel information**

When a passenger gets on a bus at a bus stop, the passenger taps his/her CityLink card on the bus to start a bus ride. When getting off the bus, the passenger taps the CityLink card again to end a bus ride. A physical CityLink card can also be virtually stored in one or many smartphone devices. PBTS only needs to know if a smartphone device or a physical card is used to tap the CityLink card for a bus ride; the passenger is not allowed to switch between his/her physical and virtual card for the ride. PBTS tracks the bus service used, the date when the ride is made, the boarding bus stop and the alighting bus stop as well as the boarding time and the alighting time of each bus ride. It is not possible for the same CityLink card (regardless of either physical or virtual modes) to be used at the exact same time (i.e., date and boarding time) to board two different bus rides; of course, rides recorded by different CityLink cards might share the same date and boarding time.

For every pair of bus stops, a base fee is stored in the database. Note that the base fee between the same two stops is not necessarily the same for both directions of travel between them. The charged fare is determined by the base fee for the specific pair of stops minus the discount (if any) provided for the specific CityLink card type used. For each bus ride, the charged fare needs to be captured in the database. Some passengers forget to tap the CityLink card when alighting from the bus; in such a case, the alight stop as well as the fare will not be recorded.

## **2.5. Function 5: Patrol operation information**

To ensure the passengers pay the correct amount of fare, patrol officers are deployed to get on a bus and validate if all passengers on the bus have correctly tapped their cards when they board the bus. The information of a patrol officer includes the officer ID (unique), name, and the number of years he/she has been employed. In case an offense is found, PBTS keeps an

offense record which includes the offence ID (unique), the patrol officer who caught the offence, the offender's NRIC, a CityLink card that belongs to the offender (if he/she has any with him/her), the bus service, the bus, the driver, the date, the time of the offense, and the penalty amount. If the offender does not have a CityLink card, the charges will be paid by cash. Note that the date and the time of the offense should be recorded as two different attributes.

### **3.1 Project Phases and Deliverables**

Your team is tasked to design the PBTS database that can capture the stated business rules as closely as possible. Please focus on the functions that are listed in this document **ONLY** and ignore other functions that an actual PBTS systems may have.

If there is any business rule or feature stated in this document that conflicts with those of any actual public bus transport system you know of, you should adhere to the business rules or features of this document.

All clarifications must be made using eLearn Discussion Forum (menu ribbon -> Discussions). Post your queries in appropriate Topics (Project Phase 1, and Project Phase 2).

It is **mandatory** for all teams to read the discussion forum on project clarifications posted on eLearn. Posts asking questions that have already been answered shall not be answered.

The project work is divided into two phases, according to the database development process.

- Phase 1: ER Modeling (**Week 4, Sunday, Feb 6, 2022, 22:00**)
- Phase 2: Database Implementation & SQL Queries (**Week 10 Wednesday, Mar 23, 2022, 22:00**)

Please refer to individual phase handout documents for the detailed deliverables and requirements.

### **3.2 Project Submission**

For every phase and as a team, you are required to submit a softcopy of your project deliverables via SMU eLearn Assignment Dropbox.

Penalty for late submission is as follows:

<b>No. of Hours</b>	<b>Marks Deductions</b>
<6 hours	10%
>6 – 24 hours	20%
>24 hours	50 %

The page limit of the submission (to be listed in each phase's handout document) is firm. For example, if the limit is 4 pages, only the first 4 pages of the report will be marked.

### **3.3 Plagiarism**

Plagiarism is strictly not tolerated, and plagiarism cases will be referred to the university's disciplinary committee.

### **3.4 Peer Evaluation and Instructor Adjustment**

During week 9, you can (optionally) complete a confidential peer evaluation of your fellow team members. However, if you have any issues within your team, do highlight to your instructors early and do not wait until the peer evaluation.

We will use these peer evaluations to make a maximal plus or minus 30% for adjustment of individual student marks over your group's grade. Nevertheless, we will not hesitate to make adjustments of more than 30% for any non-participating members of the team.