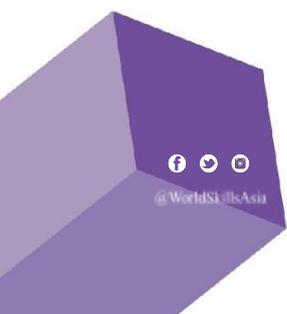


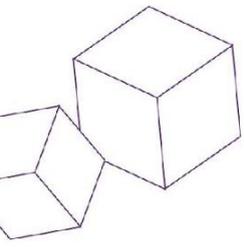
WorldSkills Asia Online Friendly Skills Games 2021

IT Software solutions for business

TEST PROJECT

Session 2





Database and import

A database for the current session is provided in script format (mssql and mysql). Structure of the data is also described in the provided **DataDictionary**. Please, **don't change** the structure of the database, because it is important for the import task and assessment.

To ensure the initial filling of the database, the ability to test the application and the convenience of evaluation, import data from the provided files (the "Import" folder) into the database you created. Pay special attention to preparing the data for import.

Desktop

It is necessary to implement part of the functionality for employees of parking centers.

It is necessary to take into account the provided application layouts and arrange the elements in accordance with the layout.

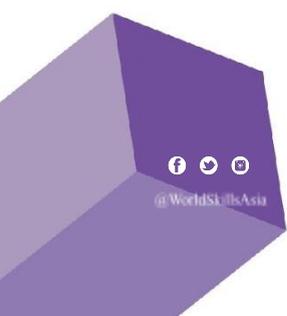
The following important functions are provided within the desktop application:

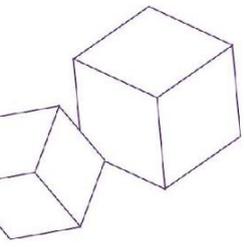
- view the list of cars with the functions of searching, filtering and sorting data;
- view detailed information about the car with a list of drivers, the history of actions during the current trip and the history of parking payments;
- monitoring of car parking both on the diagram and on the video.

To work with a desktop application in the current session, you are provided the API and documentation (as postman collection and as html document), how to use it:

- API method for calculating the distance by coordinates of two points,
- API method for getting the current location of the car,
- API method for getting video from parking surveillance cameras.

In most screens, it is necessary to provide periodic updating of information. If you have difficulties with the implementation of the update on the time interval, then provide an additional button that should update the required information.





Home Screen

The application must work in one window and have 2 main parts: common header part (visible for each content page) and content part, representing as tabs.

The following elements should be present on the main window (common header):

- Company logo;
- Menu in the form of three icons:
 - list of trucks,
 - video of parking,
 - parking scheme;
- Drop-down list with all existing truck parks of the company (from the database);
- The total number of parking places in the selected Truck park;
- The current number of trucks in the parking lot (update every 10 seconds);
- The number of free parking places at the moment (update every 10 seconds);
- The current date in the format "5 Aug 2021" (in the right up corner).

Further work with the application should be prohibited, if there is no truck park selected.

The remaining part of the screen should be occupied by open pages tabs. If the user tries to open a page that already exists in the open tabs list, then you just need to move the user to an existing tab, and not duplicate it.

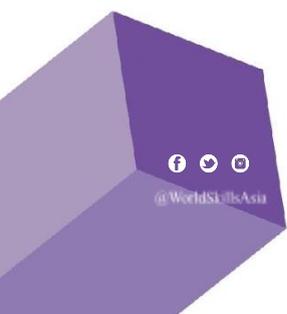
Tabs

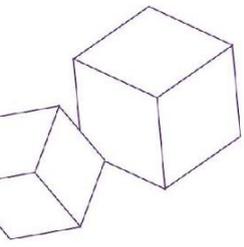
Different types of pages can be used as content for tabs:

Page Type	Heading
Page of a specific truck (truck)	Number of the truck (license plate number of the vehicle)
List of all trucks (truck list)	TRUCKS
Parking scheme	PARKING "Parking Name"
Video stream from parking cameras	VIDEO "Parking Name"

All three icons in the menu should open the corresponding type of tab.

By default, when the application is launched, a single tab with a list of trucks should be opened. Additional tabs should be added to the right one after the other. The maximum number of tabs is not limited. Each tab can be closed (for example, a cross on the right).





Truck list

It is necessary to implement functionality for displaying, filtering and searching for trucks.

Please note that the corresponding API methods are provided to get the current location of the truck and to calculate the distance to the selected parking center.

For each truck, you need to output the following information:

- identification number,
- vendor,
- model,
- type image with a text hint (type name) when hovering the mouse cursor (tooltip),
- list of drivers (separated by commas or with a line break),
- the status of the truck (on the way, in the parking lot),
- if the truck is in the parking lot at the moment, then information about the specific parking lot and parking space,
- the distance from the current location of the truck to a specific parking center (or the inscription "no info").

It is necessary to provide **filtering** of the list:

- by truck type (with a drop-down list from the database and the default value of "All"),
- by presence in the parking lot,
- by the distance from the parking center (in two ways: by manually entering the minimum and maximum boundaries, as well as using a two-way slider in accordance with the layout).

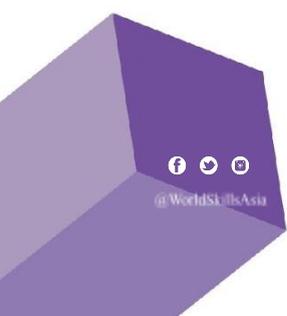
It is necessary to provide **searching** possibilities by many fields, such as vendor, model, number or drivers.

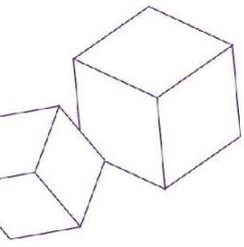
It is necessary to provide list **sorting**:

- by truck vendor,
- by distance to the parking center.

The background of the rows in the table with trucks should alternate in color.

The list of vehicles must be updated every 10 seconds to display up-to-date data on the location of the truck. When updating, the sorting and filtering parameters should not be reset. Filtering, searching and sorting should work simultaneously and consider all the parameters entered by the user.





Parking

Any parking lot has a strict structure in the grid form, where the columns are indicated by Latin letters from A to Z, and specific places are numbers starting from 1. For example, there may be parking spaces F6, R71, etc.

The maximum parking dimensions are defined in advance in the characteristics of the truck park as the number of columns and rows. Schematically, parking spaces should look like separate squares with the following possible signs:

Value	Square Border Color	Title on the square
The place is free	Green	The number of the parking space (for example, F6)
The place is reserved	Blue	R
The place is occupied	Black	Truck type image

A parking space is reserved when the start time of parking is greater than the current one (so, it is planned in the future). Only parking spaces that start within the next 3 hours should be displayed as reserved. Parking space statuses should be updated every 10 seconds.

The user should be able to register new parking using right block, that contain:

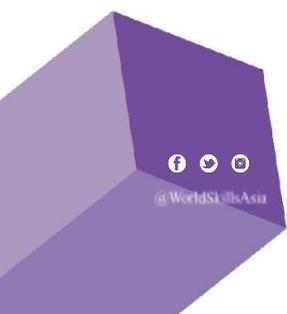
- truck (the selection list should only include trucks that are located within a radius of 100 km from the current parking center),
- parking space (there should be only free places in the selection list),
- the estimated date and time of departure from the parking lot (strictly more than the current time).

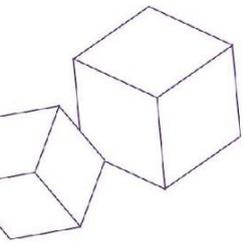
The parking place should be filled automatically, when the user clicks on the free parking rectangle. But if the user clicks on a reserved or occupied parking place - the corresponding message should be shown and nothing selected in the combo box.

All fields are required and the saving button should be disabled without filling it.

The system should check new long-term parkings with future reservations and prohibit new parking with a corresponding message if they overlap.

After confirmation, the data should be saved in the database, and the parking scheme should be updated taking into account the occupied space.





Video stream

Employees of the parking center should be able to monitor the parking lot using video cameras. The data from the cameras can be obtained using the appropriate API method. In the rectangle with the camera, you need to display an html-page in which the “video” from the camera will be automatically displayed.

On the screen, you need to display video from all the cameras installed in the parking lot, in accordance with the layout and simultaneously. For each camera, the number and name are displayed in the format "cam #{ParkingNumber} - {ParkingName}". Since there may be situations when the camera is temporarily disabled, it is necessary to display its status:

- green circle in front of the name - the camera is working,
- red circle - the camera does not work.

Truck card

You can move to the truck page from the list of trucks. The truck page consists of the following main blocks:

1. Basic information,
2. Drivers,
3. History of actions related to the truck.

Basic information

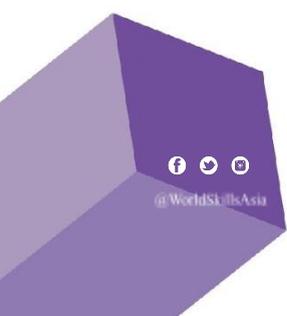
It contains the truck type in the form of an image, as well as the brand, model and ID-number of the truck.

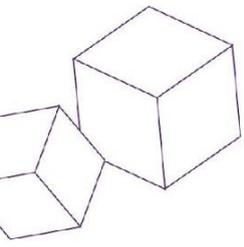
Drivers

The list of drivers is displayed as cards. The driver's card must contain:

- photo,
- name,
- date of birth,
- the number of trucks that can be driven,
- phone number (it should be possible to select and copy the phone number).

If the truck is currently on the way, then for such a driver the card should be displayed with a green background; otherwise, with a gray background.





Action history

The list should display road actions within the all routes for the current truck, such as entry/exit from the parking lot, health check, driver change, repair, and others. All actions should be ordered by date and time from newest to oldest (the most recent at the top).

For each action the type, date and time are necessarily displayed (time in 24h format), as well as colors (background and font) that depend on the type of action, visually corresponding to the type of action. As additional information (if it is present in the database), the truck park and the parking place number, as well as the driver's name can be displayed with the following special rules:

- if any road action contains information about driver - display his full name;
- if road action is driver changing - display both drivers (at first who **was** driving, then who **will be** driving), separated ' -> ';
- for the parking it is necessary to display enter and exit actions with park truck name and parking space number.

Keep it in mind that all road actions **except parking** are stored at the same table in the database and have corresponding types and colors. Detailed information about parkings is stored in the specific table and should display in gray color (without special type color in database).

