**UTCN**
**Computer Science Department**
**Software Design 2019**

# ASSIGNMENT A3

==========

## 1. Objective

The main objective of this assignment is to allow students to become familiar with the **client-server** architectural pattern.

## 2. Application Description

Use JAVA/C# to design and implement an application for an online Parking Request System. The main goal of the application is to provide clear and transparent way of requesting and assigning parking spots in the parking lots of Cluj-Napoca. Each citizen can make a new request for a parking spot. The request can be done only for one car that the citizen owns, but he can select multiple parking lots that would suit him good. Each parking lot has a certain number of parking spots. If the citizen has multiple cars, he must file multiple parking requests.

The application should have two types of users: a regular user represented by the citizen and an administrator user, represented by the city clerk responsible for assigning the free parking spots. Both kinds of uses have to provide an email and a password in order to access the application.

The citizen (regular user) can perform the following operations:

- Register a new account.
- Add/Remove owned Car.
- View all request made by him.
- Make a new request.
- Update/Delete request.

The clerk (administrator) user can perform the following operations:

- See list of parking lots.
- See a list of parking requests for a parking lot (ordered by request date).
- Assign Parking Spots to citizens. **After the parking spot is assigned, the citizen who made the request receives a notification.**
- Retract Parking Spots from citizens who did not pay the parking tax.
  (Mark parking spot as free)

## 3. Application Constraints / Requirements

- Implement the client-server communication, for the following features:
    - Login
    - User creates a parking request.
    - Clerk sees list of active requests for a parking lot.
    - Clerk assigns parking spot.
    - User receives notification when parking spot is assigned. (Alternative: you can notify **all** the users that a parking spot has been assigned to someone, not necessarily them)

- Use TCP sockets and JSON serialization to send data between the client and the server.

- Use the Observable Design Pattern to implement the User notification. The notification is a UI dialog showing a short message (e.g."*You have been assigned a parking spot*").

- All necessary auxiliary data can be inserted in the DB before the application starts and hardcoded in the UI. You do not need to implement logic for creating/retrieving Parking Lots, Parking Spots, Cars of a user or any other data you may need to post a request and assign a parking spot.

- All other features can remain mocked or be skipped if not yet implemented. For example, after the login, the admin can be directed to the view with parking requests for a specific -*hardcoded*- parking lot, if the list of parking lots has not been already implemented.

## 4. Grading

| Grade | Functionality |
|---|---|
| **5** | Client-server connection for:<br>● Login<br>● Create Request (Client) |
| **6** | Client-server connection for:<br>● See list of Requests for a parking lot (Admin) |
| **8** | Assign parking spot (admin) + Notification for User |
| **10** | Analysis and Design Document |

**Bonus points:**

1. **Additionally to the notification in the application, send an Email notification when parking spot is assigned (1p)**
2. **Full functionality with client-server communication (2p)**
   - **2.1. Register user (0.25p)**
   - **2.2. See List of Requests - User (0.25p)**
   - **2.3. Add Car (0.25p)**
   - **2.4. Remove Car (0.25p)**
   - **2.5. Update Request (0.25p)**
   - **2.6. Delete Request (0.25p)**
   - **2.7. See List of Parking Lots (0.25p)**
   - **2.8. Retract Parking Spot (0.25p)**