PublicTransport

1. Introduction

At this moment, in my city (Brasov city- Romania country), in every public station there is a schedule for busses. Based on this schedule a commuter can find out when the next bus comes in a station. All this information is also available on the public transport website.

Features:

Because the main goal for this application is to come in hand for commuters, these features are the minimal requirements in order to minimise the time spent in stations.

a) Update the schedules daily
b) Allow the user to visualise the bus schedules, in an easy to read format.
c) Allow the user to search for a bus or station
d) Allow the user to view bus routes based on the starting and destination stations.
e) Show the nearest stations and their bus schedules based on the user current location

There are times when the public transport company, in assosiacion with commuters change routes, station names or schedules. In order to provide the most accurate information, the mobile application should update this information daily.

Viewing the bus schedules in an easy to read format comes in hand when deciding which bus to choose. If there is no bus nearby that fits the end-user requirements, he can search for a station from which he can get to his destination or search for the destination station and find out which bus can take him there.

Searching for busses which gets one user from station X to station Y will come in hand, specially for tourists which know the starting destination and where they want to go. Also, allowing the app to use a device’s location is one of the best ways to check if there’s a bus station nearby from which you can go to your destination.

2. Design and Implementation

Because I am using data from the public transport company in my city I was required to sign an intellectual property contract with the company. This means that the server side code is in their property, even though i have developed it, which means that i cannot give any information about it (except for the url endpoints which have been provided in the previous weeks.)

Regarding the client side, I used ionic in combination with some cordova plugins, as follows:

Every page is implemented on its own, for example, the stations page has a different .html page, a different controller and a separate service which comunicates with the api.

The html page:

```
<ion-view view-title="Stations">
  <ion-nav-buttons side="secondary">
    <button class="button button-icon icon ion-ios-search-strong" ng-click="showFilterBar()">
    </button>
  </ion-nav-buttons>
  <ion-content overflow-scroll="true">
    <div id="map" data-tap-disabled="true" style="display:none"></div>
    <ion-list>
      <ion-item ng-repeat="statie in statii" ng-href="#/app/ruteStatie/{{statie}}">
        {{statie}}
      </ion-item>
    </ion-list>
  </ion-content>
</ion-view>
```

The controller:

```
.controller('StatieController', ['$scope', '$stateParams', 'statii', 'closestRoutesFactory',
'$ionicFilterBar', '$ionicPlatform', '$ionicLoading',
  function($scope, $stateParams, statii, closestRoutesFactory, $ionicFilterBar, $ionicPlatform,
$ionicLoading) {
    $ionicPlatform.ready(function() {
      $ionicLoading.show({
        template: '<ion-spinner icon="bubbles"></ion-spinner><br/>Acquiring data!'
      });
      statii.$promise.then(function(response) {
        $scope.statii = response;
      }).finally(function() {
        $ionicLoading.hide();
      });
    })
    $scope.showFilterBar = function() {
      var filterBarInstance = $ionicFilterBar.show({
        cancelText: "<i class='ion-ios-close-outline'></i>",
        items: $scope.statii,
        update: function(filteredItems, filterText) {
          if (filterText == null || filterText == "") {
            $scope.statii = statii;
          } else {
            var statiiGasite = [];
            for (i = 0; i < statii.length; i++) {
              if (statii[i].toLowerCase().indexOf(filterText.toLowerCase()) > -1) {
                statiiGasite.push(statii[i]);
              }
            }
            $scope.statii = statiiGasite;
          }
        }
      });
    };
  }
])
```

And the service

```
.factory('statiiFactory', ['$resource', 'baseURL', '$http', function($resource, baseURL, $http) {
  var factory = {};
  factory.getStatii = function() {
    return $resource(baseURL + "statii", null, {
      'update': {
        method: 'PUT'
      }
    });
  };
  return factory;
}])
```

All other pages follow the same format.

Based on the design specified in a previous week (also available here: https://pidoco.com/rabbit/api/prototypes/190498/pages/page452296736.xhtml?api_key=UZTZza2GSQJyg0fWN858NIq6xRn64GOg1jD4G1H4&mode=plain) there were no changes needed.

Because the design is simple and the targeted design was set from the beginning, the final result looks exactly as explained above.

Plugins used:

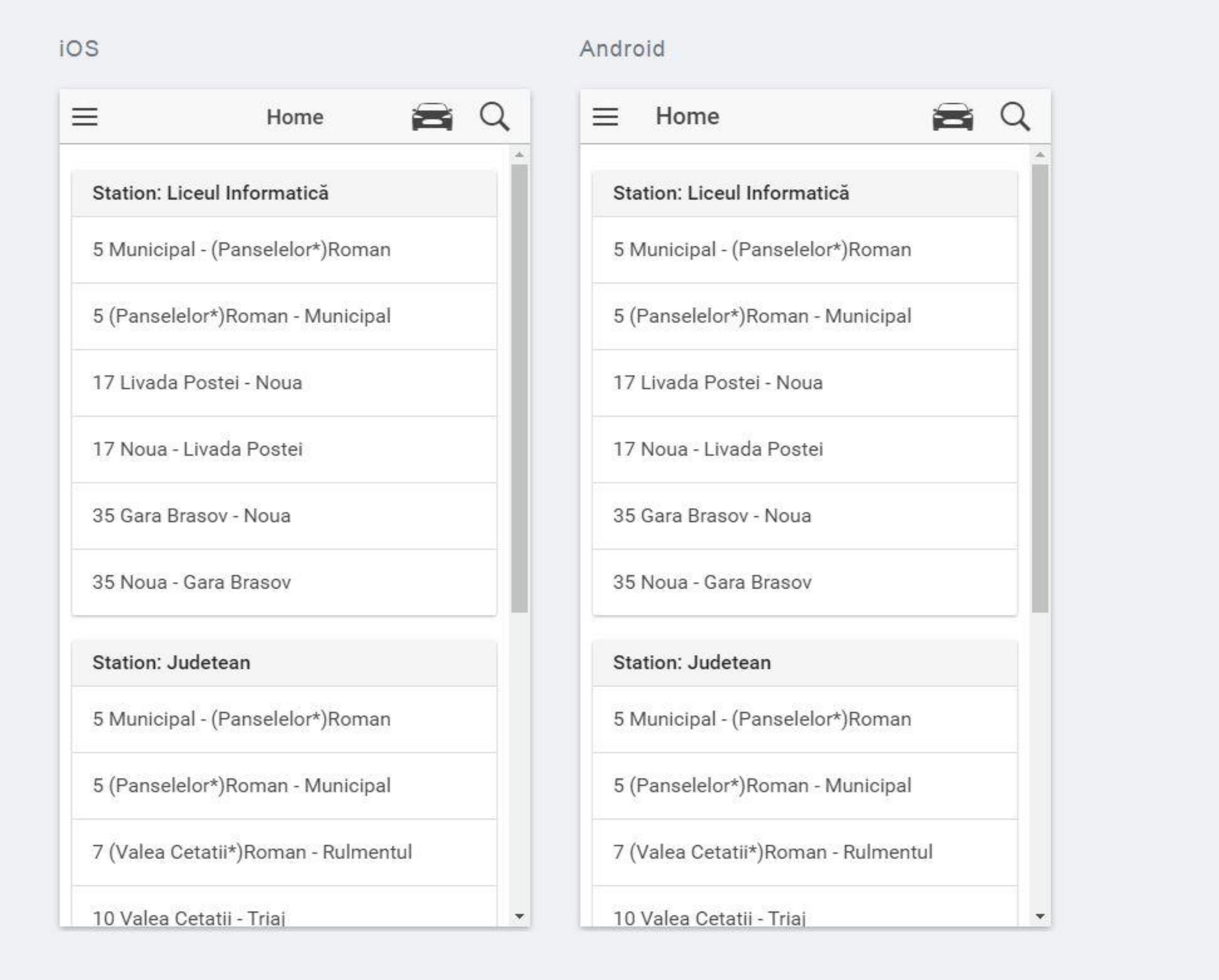
Because I wanted to provide the user with the stations available in his proximity I have used cordovaGeolocation plugin (found here: https://cordova.apache.org/docs/en/latest/reference/cordova-plugin-geolocation/). I used this plugin because it has support for all major mobile platforms.

Based on this, I used Google maps api to get the closest stations based on the location provided by the geolocation plugin. I have used Google because Bing doesn't have public transport data available for my city.

No other plugins were needed to develop this application.

Application screenshots:

Home page (which includes the closest stations based on the user's proximity):



iOS

Home

Station: Liceul Informatică

5 Municipal - (Panselelor*)Roman

5 (Panselelor*)Roman - Municipal

17 Livada Postei - Noua

17 Noua - Livada Postei

35 Gara Brasov - Noua

35 Noua - Gara Brasov

Station: Judetean

5 Municipal - (Panselelor*)Roman

5 (Panselelor*)Roman - Municipal

7 (Valea Cetatii*)Roman - Rulmentul

10 Valea Cetatii - Triaj

Android

Home

Station: Liceul Informatică

5 Municipal - (Panselelor*)Roman

5 (Panselelor*)Roman - Municipal

17 Livada Postei - Noua

17 Noua - Livada Postei

35 Gara Brasov - Noua

35 Noua - Gara Brasov

Station: Judetean

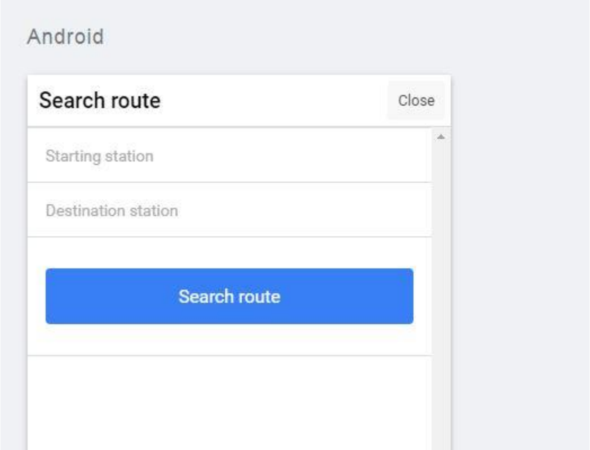
5 Municipal - (Panselelor*)Roman

5 (Panselelor*)Roman - Municipal

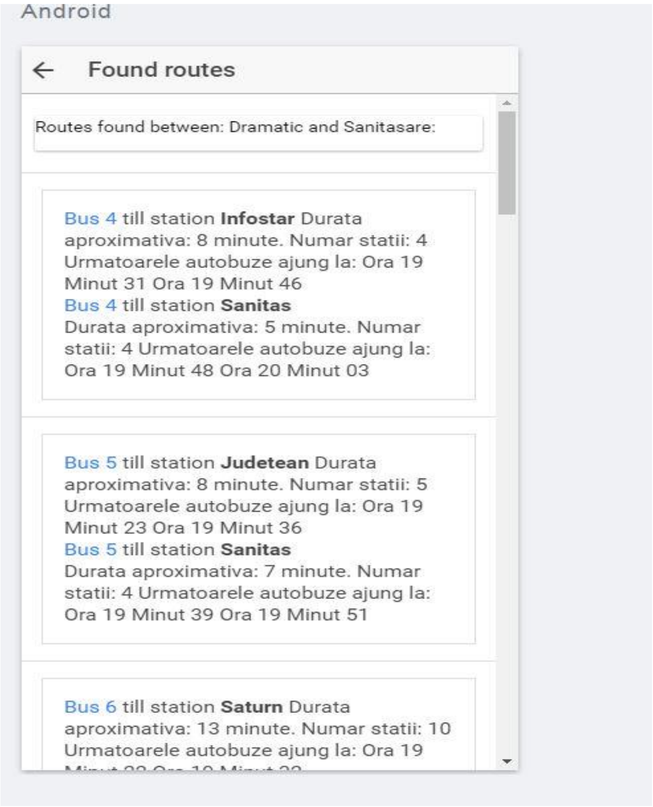
7 (Valea Cetatii*)Roman - Rulmentul

10 Valea Cetatii - Triaj

Searching for routes:



Search routes results: (ignore the language mixups)



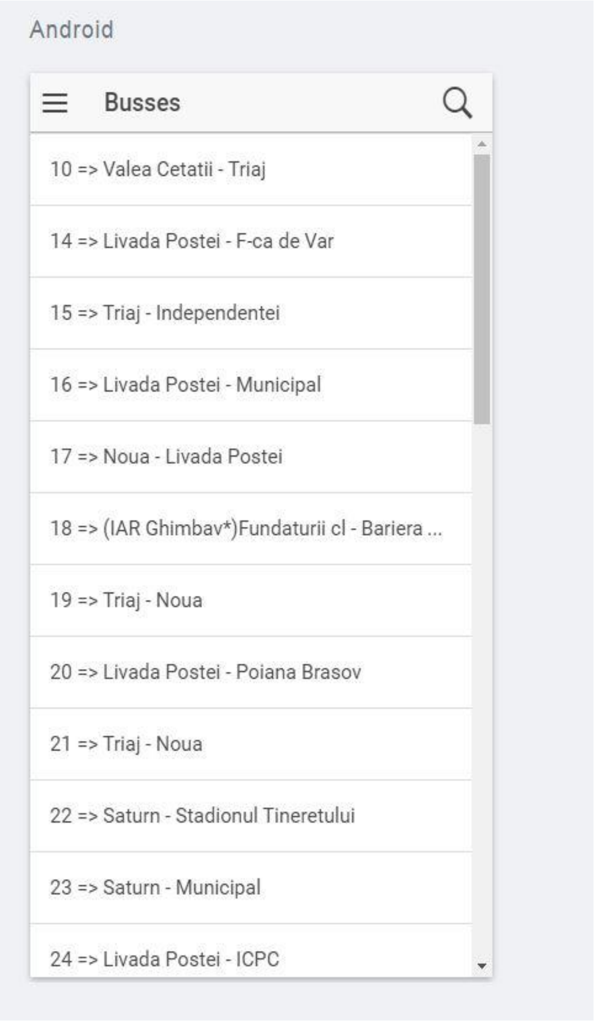
Routes found between: Dramatic and Sanitasare:

Bus 4 till station **Infostar** Durata aproximativa: 8 minute. Numar statii: 4 Urmatoarele autobuze ajung la: Ora 19 Minut 31 Ora 19 Minut 46
Bus 4 till station **Sanitas**
Durata aproximativa: 5 minute. Numar statii: 4 Urmatoarele autobuze ajung la: Ora 19 Minut 48 Ora 20 Minut 03

Bus 5 till station **Judetean** Durata aproximativa: 8 minute. Numar statii: 5 Urmatoarele autobuze ajung la: Ora 19 Minut 23 Ora 19 Minut 36
Bus 5 till station **Sanitas**
Durata aproximativa: 7 minute. Numar statii: 4 Urmatoarele autobuze ajung la: Ora 19 Minut 39 Ora 19 Minut 51

Bus 6 till station **Saturn** Durata aproximativa: 13 minute. Numar statii: 10 Urmatoarele autobuze ajung la: Ora 19 Minut [illegible] Ora 19 Minut [illegible]

Busses:



Android

Busses

10 => Valea Cetatii - Triaj

14 => Livada Postei - F-ca de Var

15 => Triaj - Independentei

16 => Livada Postei - Municipal

17 => Noua - Livada Postei

18 => (IAR Ghimbav*)Fundaturii cl - Bariera ...

19 => Triaj - Noua

20 => Livada Postei - Poiana Brasov

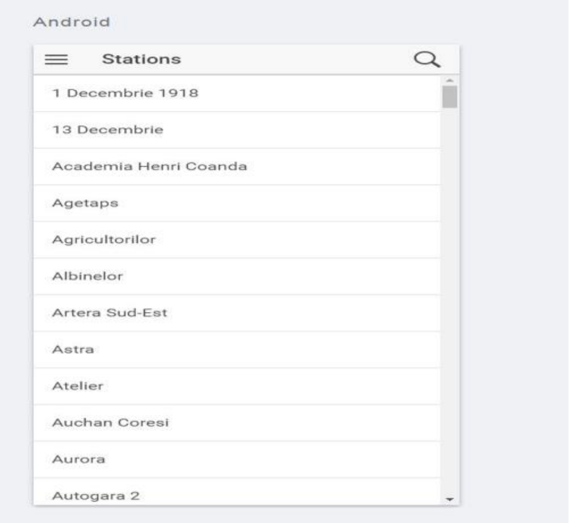
21 => Triaj - Noua

22 => Saturn - Stadionul Tineretului

23 => Saturn - Municipal

24 => Livada Postei - ICPC

Stations:



Android

Stations

1 Decembrie 1918

13 Decembrie

Academia Henri Coanda

Agetaps

Agricultorilor

Albinelor

Artera Sud-Est

Astra

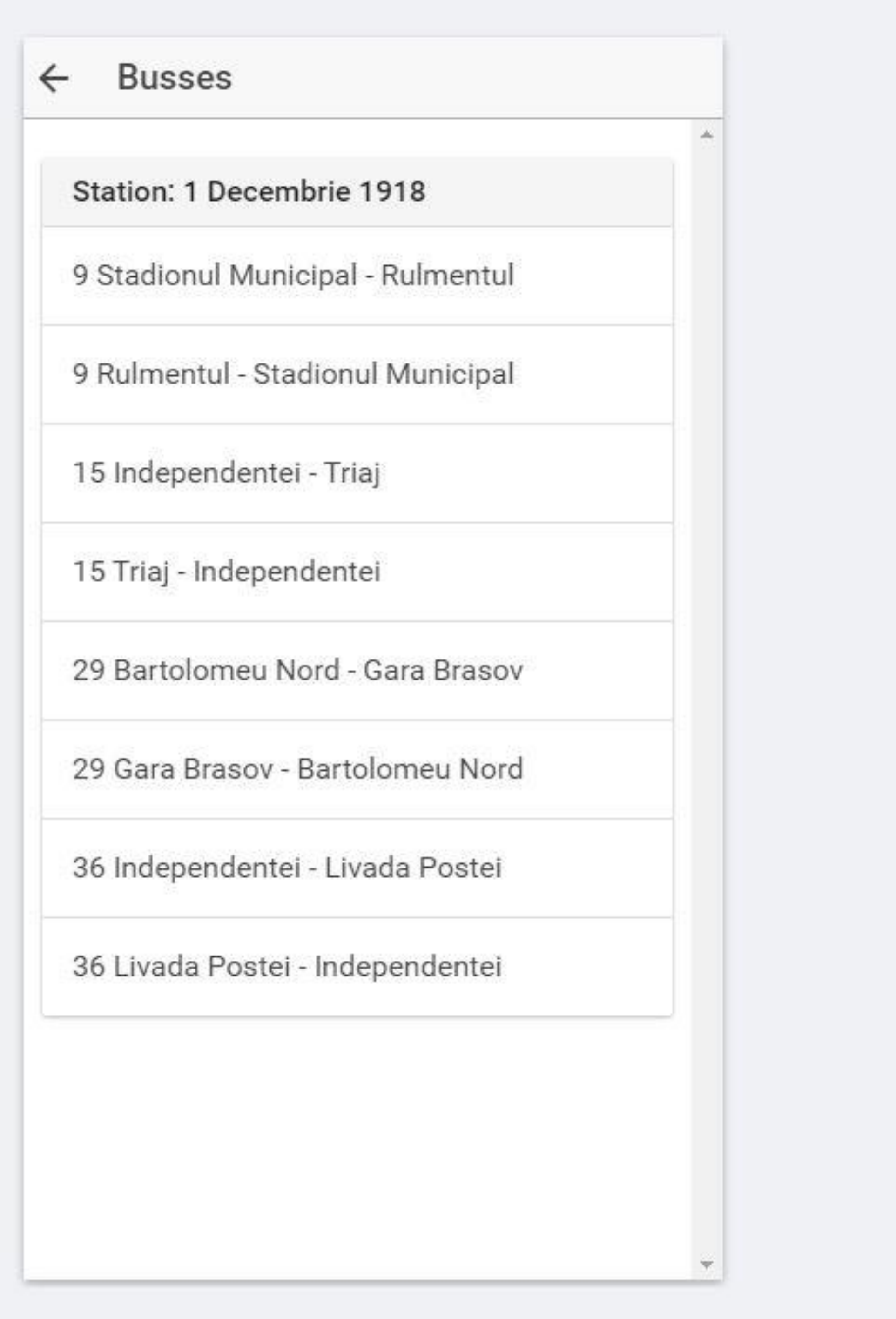
Atelier

Auchan Coresi

Aurora

Autogara 2

Busses which go through station:



Station: 1 Decembrie 1918

9 Stadionul Municipal - Rulmentul

9 Rulmentul - Stadionul Municipal

15 Independentei - Triaj

15 Triaj - Independentei

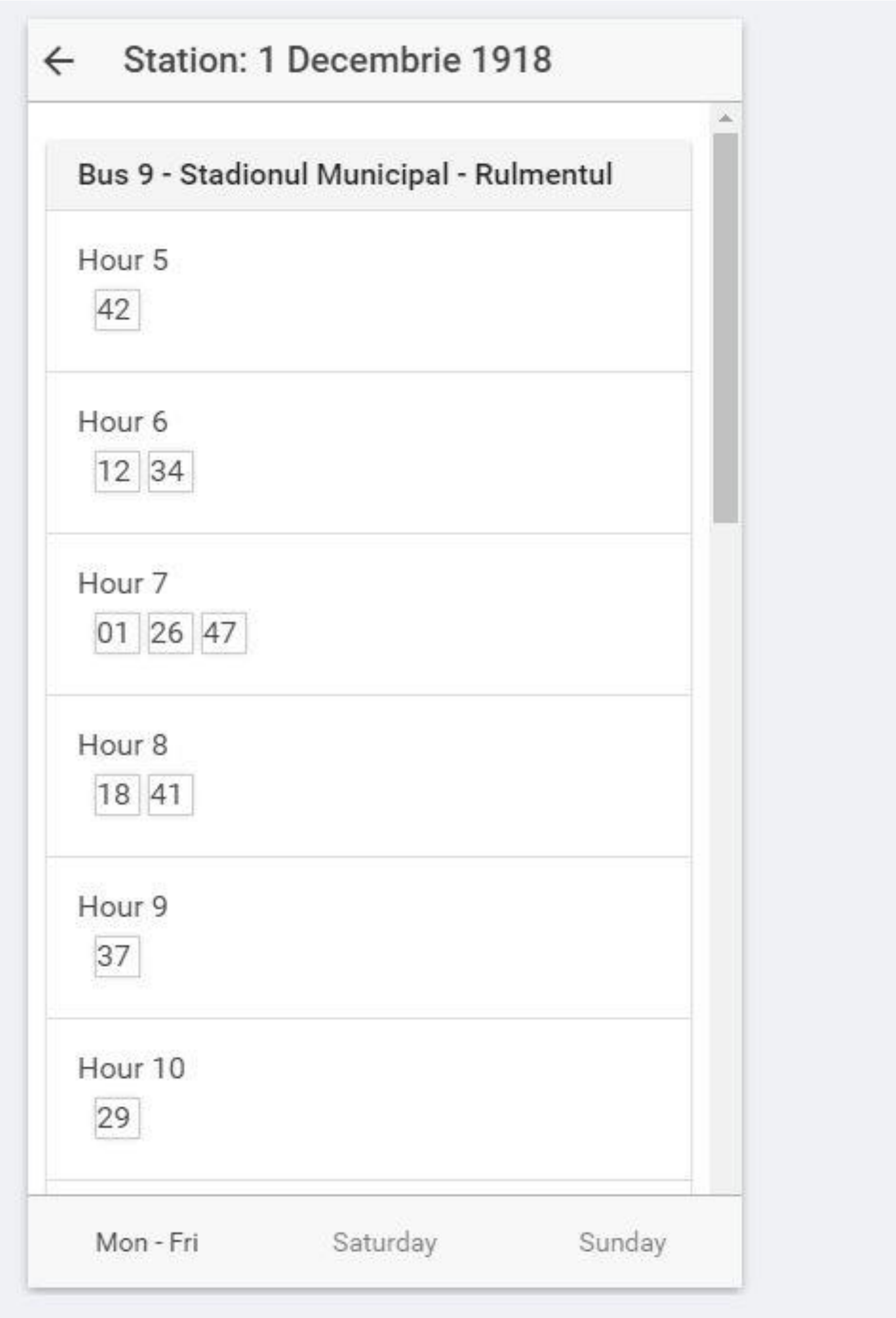
29 Bartolomeu Nord - Gara Brasov

29 Gara Brasov - Bartolomeu Nord

36 Independentei - Livada Postei

36 Livada Postei - Independentei

Buss schedule for a station:



Station: 1 Decembrie 1918

Bus 9 - Stadionul Municipal - Rulmentul

Hour 5
42

Hour 6
12 34

Hour 7
01 26 47

Hour 8
18 41

Hour 9
37

Hour 10
29

3. Conclusions

By working on this project I provided a new and easy way for commuters and tourists in my city to use the public transport.

Upcoming features:

One of the most important features (which is already in progress as you can see in the source code) is the ticketing and subscription support. The next step is to work with the public transport company to allow NFC payments directly from commuter's phones.

Different choices:

One on the problems which I encountered in this project is the public transport company strict server restrictions and it became a life lesson. Do not work with public companies for

As of now, the server is unavailable based on security issues which they have to adress.

4. References

Public transport company page: http://ratbv.ro/