1. Why should you solve this assignment?

Solving this assignment is beneficial for you, because:

- we can assess your current technical skills and experience and take them into consideration when setting the Bootcamp training pace;
- you get the experience of developing a small information system using contemporary tools and technologies.

This assignment has been created with the assumption that you should be able to solve at least some parts (stages) of it. Our advice is to try and **solve it step by step**, and **submit your work even if you couldn't solve the whole assignment**. Last year, most candidates who solved and submitted about half of the assignment, made it to the next round.

A junior developer who has completed an IT/development curriculum at IT College, the Tallinn University of Technology or a similar institution of higher education, should be able to solve the whole assignment in 3-5 days, assuming they are familiar with the technology or bring themselves up to speed quickly.

Some stages of the assignment can be solved by studying the attached examples and modifying the existing solution. Other functionalities require independent study and experimenting.

The deadline for submitting the assignment is **26th March** (incl).

In order to solve the assignment, you need to know the basics of:

- the programming language you are going to use (Java, .NET C# , PHP, Progress)
- web technologies (HTML, CSS)
- database queries and design

You will also have to understand (based on the existing solution and possibly some independent study) the following concepts:
1) the Model-View-Controller (MVC) design pattern;
2) the ORM (Object-Relational Mapping) technique;
    a. for .NET, the Entity Framework;
    b.for Java, the Hibernate framework;
    c. for PHP, the Doctrine framework.

2. How does Helmes grade the assignment?

a) higher score for a reliable solution (the solution works and does not produce technical errors);
b) higher score for a correct solution (the solution works the way it is intended/described in the assignment);
c) higher score for clean code and good technical design.

You could get **bonus points** for:
a) a good user interface design (the UI is visually pleasing, simple and easy to use)
- you can use ready-made templates and standard development tools
- responsive design is a plus

b) an error-proof and secure solution.

3. How to get going?

Read the assignment and pick the technology you're going to use to solve the assignment, then follow the relevant link to the assignment materials. The materials under the link include a tutorial for setting up the development environment and links for further reading that should help you in solving the assignment.

.NET C# technical assignment - http://bit.ly/2mSi6XX
- „Bootcamp NET - setup tutorial.pdf"
- „helmes-bootcamp-net.zip"

Java technical assignment - http://bit.ly/2nfDirG
- „Bootcamp Java - setup tutorial.pdf"
- „helmes-bootcamp-java.zip"

PHP technical assignment - http://bit.ly/2nfEew7
- „Bootcamp PHP – setup tutorial.pdf"
- „helmes-bootcamp-php.zip"

The necessary materials for the **Progress** language are available on demand.

Should you run into problems that you can not overcome while setting up the environment, we can provide technical support - to a reasonable extent (see pt 6, Technical support).

4. Assignment background information

4.1. Current situation

Your client, Gregor, has a business, called Tyres24 that offers tyre exchange services in Tallinn. Tyres24 started out at a garage in Nõmme and recently bought two more garages: one in Lasnamäe and the other in Mustamäe.

The Nõmme and Mustamäe garages are larger and also offer tyre storage service ("tyre hotels"). These two garages have two service lines: one is for cars only, the other line can be used for servicing cars, trucks or vans. This means that these garages can service two cars or one car and one truck/van at the same time.

The third garage in Lasnamäe is smaller and does not have a tyre hotel; only one car can be serviced at a time and van or truck tyres can't be exchanged there.

Customers can book time slots for tyre exchange by phone: each garage has a general phone that employees pick up if they're free at the moment: there are no secretaries or administrators. Bookings are written down into a paper notebook by hand.

Each garage can only see their own bookings and if the time slot is taken, they can't tell if the other garages could take the customer, instead. Also, because bookings are written by hand, some important information (e.g. the customer's phone or the car's licence plate number) may be missing or very difficult to understand. If the owner wants to know how his garages are doing, he has to retrieve all the notebooks and go through them manually; since this takes a lot of effort, he only does this a few times a year.

The garages are staffed by employees aged 18 – 40. The number of staff at a garage fluctuates: during busier periods in winter, the larger garages each may have up to 5 employees present; in summer, only one person may be working at the garage. The smaller garage may have 1-3 employees present at a time. The planning of employees' work schedules is another area the owner of the business would like the new system to be of help.

4.2. Client's needs

The first thing the client wants is for employees at the garages to be able to enter bookings into the system so that all bookings have all the necessary data and the information is legible. Another benefit of the system would be to have an up-to-date overview of bookings at all the garages so that if one garage is busy, they could refer the client to another garage of the same company.

Allowing customers to book their tyre exchange by themselves would lessen the burden of picking up the phone and taking bookings for garage employees and also be more convenient for the customers.

The owner expects the system to enable him to get statistical information on the bookings at each garage quickly and effortlessly so he can plan employees' schedules better.

The system's users will be the garage staff (to enter and view bookings), the owner (for viewing booking statistics) and the customer (to book a tyre exchange). The system should be user-friendly and intuitive for all user groups.
The client is not too concerned about data security, so authentication and authorisation is not required.

4.3. System development roadmap

The system should allow the user to add, edit, view and erase the data. It will be accessed using the Chrome browser on Windows computers. Responsive design is not mandatory, but will add bonus points. There are no restrictions to the realisation: using templates, standard development tools and frameworks (e.g. Bootstrap), etc is allowed.

The system will be developed in stages in such a way that once a stage is ready, it can be used by the client. Each stage adds functionality to the system, making it more valuable for the client.

At the moment, Gregor's son has realized "stage-zero" of the system that allows the user to enter tyre exchange start time into one field and the rest of the booking information into another and save the booking. The bookings can also be viewed as a list with the latest added bookings at the top.

Since his son no longer has time to work on the system, this will be the starting point for your assignment and has been included in the assignment materials (see **pt 3**).

5. Your assignment

We suggest you solve the assignment stage by stage: earlier stages are easier and you might need to complete the previous stage to solve the next one; later stages may be technically more complex or more time-consuming.

However, if you think you can't solve a stage, try don't hesitate to try later stages.**If you manage to solve only a part of a stage, submit the partial solution. You don't have to complete a stage to move on to the next.**

It is up to you how many stages and how much functionality you decide to develop, but you do have to keep to the deadline. **It is perfectly OK to submit half of the assignment.**

Keep track (e.g. in an Excel or text file) of how long you spent:

- setting up the development environment;
- researching, looking for information, studying;
- programming and solving related technical problems;
- testing.

NB! Since we are using automated tests to validate your assignments, please **add HTML ID fields to all buttons and inputs using the following rules**:

- All lower-case dash-case names (e.g dash-case)
- Use button/link/input name/label as ID (eg New Booking becomes new-booking)

When listing entries, use database entity name as first part and append dash with ID. So when you are listing bookings (which is booking in the database) links to booking details should be booking-1,..booking-N)

Stage I

As a result of the first stage, the user (a Tyres24 employee) should be able to enter (not edit or delete) a booking and view a list of bookings and booking detail information.

I.a. Booking form

Develop a data entry form for the garage employee to enter a booking. The form must have the following fields:

- Garage
- Type of vehicle (car, van, truck)
- Start date and time
- End time
- Licence plate number
- Customer's phone number
- Tyre hotel - indicating if the customer wants to use the 'tyre hotel' storage service
- Additional comments

The system does not have to check if all the fields have been filled in and are in the correct format and does not have to calculate the end time: the user is responsible for that.

I.b. List view

Develop a view where the user can see all bookings in all three garages, sorted by start time (the bookings starting next should be at the top of the list). The list should have the following fields: garage, start time, type of vehicle, licence plate number. The list does not have to be paginated.

By clicking on the booking, the user should be able to see booking detail view showing all booking data.

Stage II

As a result of the second stage, the system should check for missing or incorrect data on the booking form and allow the employees to edit and delete bookings.

II.a. Data validation

The system should check that:
i) all required (Garage, Type of vehicle, Start time, End time, Licence plate number, Customer's phone number) have been filled in;
ii) "Garage" field can only contain garage names - possible values: 'Nõmme', 'Lasnamäe', 'Mustamäe';
iii) "Type of vehicle" can only have three values: 'Car', 'Van', 'Truck';
iv) "Start time" and "End time" must be in the DD.MM.YYYY HH:MM format (24-hour clock);
v) "Licence plate number" must have 4-9 characters and can only contain numbers (0...9) and Latin letters (A...Z);
vi) "Customer's phone number" can only contain numbers and must have at least 5 characters;
vii) "Tyre hotel" must be a Boolean type field (yes/no);
viii) "Start time" and "End time" can't be in the past;
ix) "End time" can't be earlier than "Start time" and can't be later than 2 hours from start time.

If one or more of the conditions is not met, the booking will not be saved, the user must be informed and the problematic fields highlighted.

II.b. Managing bookings

The user should be able to edit and cancel bookings. A cancelled booking must not be deleted from database but should be marked as cancelled. If a booking is edited or cancelled, the date and time of editing/cancellation must stored and displayed in booking detail view.
In list view, edited and cancelled bookings must be displayed differently from unmodified bookings:

- edited bookings must be displayed in red font
- cancelled bookings must be displayed in grey font and strikethrough

Stage III

Stage III should add the option of managing (adding, editing and deleting) garages and help the user avoid double-booking.

III.a. Managing garages

The system must allow the user to add a new garage or add a service line to a garage.

For a new garage, the following information is to be entered:

- Name of the garage (displayed in booking form as one of the options for the Garages; free text, up to 15 characters, required)
- Garage address (one field, free text, up to 200 characters, required)
- Whether the garage has tyre storage (Boolean, required)

For a new service line, the following information is to be entered:

- Service line ID (number from 1 … 10; required; must be unique in the garage)
- Whether the service line can service vans/trucks (Boolean, required)

It must also be possible to edit this information for the new and existing garages/service lines, e.g change address or add/remove service lines. The system should validate data (required fields + format) upon saving the entry. In case of error, the system should notify the user and highlight problematic fields.

When a new garage is added, it must become available for booking a tyre exchange.
When a garage's information is edited (e.g. the name is changed or service lines added/removed), the booking form must be updated, but the information on existing bookings should also be updated.

When a garage is deleted, it must not be possible to enter new bookings for the garage, but neither the garage nor its bookings should disappear from historical data.

III.b. Fewer mistakes

In order to minimize human error, the system should do the following:
i) calculate "End time" based on "Vehicle type". For a car and a van, the tyre exchange will take 30 minutes; for a truck, a 60 minute time slot must be booked;
ii) not allow the user to book a truck or van tyre exchange at a garage where there are no truck/van service lines - e.g. at the Lasnamäe garage;

iii) check that the garage or service lines won't be double-booked, taking into account the vehicle type, because only some service lines can service trucks/vans.
If a booking already exists for a time slot at the specific garage, the system should notify the user and the new booking should not be accepted.
If a booking is cancelled (see **II.b.**), its time slot becomes free for booking again.
iv) to find free time slots at other garages faster, the list view must be updated enabling the user to filter bookings by Garage, Date and Truck/Van service line.

Stage IV
Stage IV is intended to help the owner of the business get an overview of the bookings and their dynamic in time in order to make better scheduling and business decisions.

IV.a. Monthly dynamics
This report should show the monthly number of bookings at all the garages grouped by service lines and compare it to the previous month. The report should have the following fields:

- Garage (e.g. "Nõmme")
- Service Line (the Service Line No at the garage)
- Total number of bookings (for the garage and service line) for the previous month
- Total number of bookings (for the garage and service line) for the current month
- The ratio of current months bookings to previous month's bookings in per cent: (Current month/Previous month)*100

Garage	Service Line	Previous month	Current month	%
Nõmme	1	230	176	77
Nõmme	2	89	120	135
Lasnamäe	1	98	54	55
Mustamäe	1	256	298	116
Mustamäe	2	192	156	81

IV.b. Weekly bookings
This report should show daily booking statistics at all the garages, one week at a time. By default, the current week (Monday to Sunday) is displayed. The user can pick, which week to display (e.g. "W8 13.Feb - 19.Feb").

The user should also be able to choose whether to show car bookings only, van/truck bookings only or the total number of bookings; the default value is "Total".
The bookings should be grouped by garages and days and the report should have the following fields:

- Garage (e.g. "Nõmme")
- Weekday (Mon, Tue, Wed, Thu, Fri, Sat, Sun)

- Number of Car, Truck/Van or Total bookings (if none, “0”)

W10 6. Mar - 12. Mar	Total

	Mon	**Tue**	**Wed**	**Thu**	**Fri**	**Sat**	**Sun**
Nõmme	12	10	5	11	23	35	12
Lasnamäe	8	7	3	15	12	20	10
Mustamäe	16	12	8	19	22	26	29

Stage V
Stage V allows customers book their tyre exchange online as self-service. The customer can add a booking, but not edit or cancel a booking online.

V.a. Customer’s booking form
A separate booking form should be created, following these requirements:
i) the customer’s booking form should contain the same fields as listed in **I.a.**
ii) “Start time” should not be entered manually, but for example using a datepicker. The customer should either be shown only the available times at the garage they’ve chosen or the free and booked time slots should be visibly different.
iii) ”End time” should be calculated automatically as described in **III.b. i)**
iv) upon saving the booking, the system should perform the checks listed in **II.a.** If one or more of the conditions is not met, the booking must not be saved, the user must be informed and the problematic fields highlighted.
v) when the booking has been saved, the user should be notified that the booking has been successful.

V.b. Suggest another garage
i) upon saving the customer’s booking form, the system should also check for double-booking (see **pt III.b. iii**). If the time slot at the user-picked garage is not available, the system should check if any of the other garages are free for the required time. If yes, the system should notify the user, listing the garages with available time slots (e.g. “We’re sorry, but this garage has already been booked. You could book a time at the following garages: …. ”).
ii) the garage employee’s booking form **(III.b. iii**) should be updated with the garage suggestion, as well.

Stage VI
To help the employees of the garages keep track of available tyre space, garage information should be updated and another validation rule added to both employee and customer’s booking forms. The room available for tyres is counted in ‘tyre slots’. Up to 4 car/truck or van tyres can be fitted into each slot.

VI.a. Add tyre space information

Garage information (**pt III.a.**) should be updated so that for each garage offering tyre hotel services, the number of available tyre slots at the garage must be entered:

- Field name: “Tyre slots”, integer, 1… 400, required if the garage has been marked as offering the ‘’tyre hotel’ storage service.
- For existing garages (Nõmme and Mustamäe), the number of tyre slots is 200.

VI.b. Calculating available space

The following applies both to garage employee’s and customer’s booking forms:
i) each time a booking requiring tyre storage is saved into the system, the space (number of slots) available at the garage decreases:

- car and van tyres take up 1 tyre slot;
- truck tyres use two tyre slots.

Cancelling such a booking frees up booked space the same way.
ii) before accepting a booking requiring tyre storage, the system must check that there is enough available space at the garage. If not, the booking can’t be saved and the user must be notified.

Stage VII

Stage VII should help the business owner plan the work schedules better. Gregor has noticed that whenever there is heavy snowfall, the number of customers requiring tyre exchange doubles (at least), so he keeps track of the 4-day weather forecast at the Estonian Weather Service. He would like to be able to see the weather information on the reports page created in **Stage IV**.

NB! If you encountered problems in Stage IV and could not create one or either of the reports, you can still earn maximum points for this stage, provided the reports page exists and the functionalities described below have been realised.

VII.a Displaying the forecast

The reports page should contain a table showing the next four days. Depending on the weather forecast, the user can add or remove a warning icon (an exclamation mark, snowflake, etc) to note that on that day, snow is to be expected.
The system should guarantee that every day, the correct 4 days are displayed and if a warning icon has been added to a day, it will not disappear.

Tue 14.Mar	Wed 15.Mar	Thu 16.Mar	Fri 17. Mar
!		!	

VII.b. Updating the forecast

The next step is to update the forecast automatically based on Estonian Weather Service information so that Gregor does not have to do it himself.

The four-day forecast is updated daily around noon. Background information can be found at: http://www.ilmateenistus.ee/teenused/ilmainfo/eesti-prognoos-xml/ (English version: http://www.ilmateenistus.ee/teenused/ilmainfo/?lang=en) and the forecast can be accessed at: http://www.ilmateenistus.ee/ilma_andmed/xml/forecast.php?lang=eng .

If the weather forecast for Tallinn's closest weather station, Harku (the <*name*> field) shows either "Heavy snowfall"or "Heavy snow shower" (the <*phenomenon*> field), a warning icon should be displayed for the day.
It is sufficient for Gregor if the forecast is updated once a day.

6. Technical support

If you need technical support, write to: bootcamp@helmes.ee and we will provide support if:

- you need help setting up the technical environment;
- you want to clarify the assignment: you've found a human error, discrepancy or something seems unclear.

We'll try to reply within two working days.

7. How to send the solution to Helmes?

1) Compress your solution into a .zip file
2) Make sure all necessary files have been included (in the .zip file):
    a) Technical solution containing all necessary code files
    b) Database or SQL scripts
    c) If you kept track of the time spent (see **pt 5**), the time-tracking file (e.g. Excel, Word, .txt…)
3) Upload the .zip-file into your personal cloud (e.g. Dropbox, Google Drive, OneDrive)
4) Send an e-mail containing the link to your .zip-file to bootcamp@helmes.ee.