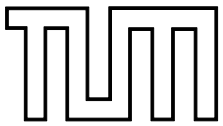


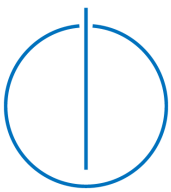
FAKULTÄT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

FleetMe Management & Maintenance

Project Report

Advanced Practical Course Automotive/Mobility Services

Baris Yazici
Diego Gaboardi
Muhammad Asad
Lukas Kick



Summary

The Car as a Service market undergoes a progressive trend in growth due to innovative technologies and higher interconnectivity in the automotive industry. By 2018, all new vehicles in Europe are obligated to be equipped with a SIM-card in order to adapt to the eCall functionality for sending emergency signals. Also, the European legislation set up a roadmap for providing the necessary infrastructure in the following years. This trend of high interconnectivity is already visible in the fleet management market.

Based on the latter, our team – Be Most Wanted – came up with an idea of making use of new services in the latter discussed telematics market. Providing fleet managers with the highest information depth while at the same time emphasizing major problems within the fleet lead to an overall simplified fleet management. Moreover, costs regarding maintenance of vehicles, fuels costs and others can be saved. The result of the project is a web platform built for fleet managers to oversee the status of the vehicles, the driving behaviour of employees and the costs of fuel amongst others. All of those features are included in a notification centre where problems based on a pre-set threshold are shown to the fleet manager. By using the BMW CarData platform, vehicles can easily be registered in the FleetMe system without the need of installing any hardware components. Following the design thinking approach, the idea of the fleet management software was iteratively adapted and renewed. In this report all steps from the ideation to the implementation of the prototype as well as the calculation of a financial business case are detailed. Thereby, the project was part of the Advanced Practical Course Automotive/Mobility Services in the winter term of 2017/2018.

Keywords: FleetMe, Fleet Management, Be Most Wanted, platform implementation, driving behaviour analysis, web service, business plan, Car as a Service, market overview.

Table of Content

1 Introduction

1.1 General Motivation

Given access to one of the most recent releases of BMW – the BMW CarData platform – the goal of each team in the practicum was to develop an innovative service that makes use of the provided data. One of the four teams is Be Most Wanted.

The BMW CarData platform is the first of its kind and was introduced by BMW in early 2017. By collecting vehicle data of BMW customers, companies can pay for using the service in order to provide new business models to their customer base. Those services are called telematics services due to the combination of telecommunication and informatics by connecting two separate information systems. By now, only few companies are already taking advantage of this state of the art technology. For example, the startup Friday is already cooperating with BMW to provide premiums to customers of vehicle insurance companies (Verischerungswirtschaft-heute, 2017). Lately, other car manufactures such as Daimler are also offering a similar telematics service for their customers. Being one of the first teams while at the same time having direct support of the development department of BMW are two determinants of potential success. Therefore, we started the course work by generating initial ideas within the team. Followed by an iterative process of improving on the initial idea, we came up with a new intriguing service within the fleet management market. Taking the opportunity by building a prototype was the logical next step. During the whole process, we were strongly supported by the chair of information systems at the Technical University of Munich and namely by our coach Max Schreieck as well as all of the lecturers.

The following success story is outlined in the next sections of this final project report.

1.2 The Team behind FleetMe: Be Most Wanted

In the early stages of venture capital funding, business angels do not only invest in the idea or the concept of the product, but also in the team that build the potential next “thing”. Also, the general idea is most often modified or even completely changed due to unknown problems or new ideas. Therefore, this section focuses on the highly diverse team that built the FleetMe Management & Maintenance service.

First of all, all team members adapt a technology driven mindset through all of us are studying in the department of informatics at the Technical University of Munich. Apart from that, each team member brought his own unique skillset and experience into the project. Therefore, four roles were defined at the beginning of the implementation phase. Starting with the implementation of the backend, Baris Yazici mostly worked on the data collection from the BMW CarData platform using NodeJS and MongoDB. Second, Diego Gaboardi built the connection of the database to the user interface so that trips of employees as well as the status of the vehicles are updated on the front ReactJS panels. The business-related activities, the implementation and the design of the user interface were conducted by Lukas Kick. At last, Muhammad Asad focused mostly on additional project activities such as the blog entries and in particular the making of the trailer of our prototype. In the following Table 1 each of the team members is shortly introduced:

Profile image	Name	Project contributions
	Baris Yazici (Computer Science)	> Developed the backend server including the database and connection to BMW CarData > Set-up of dummy data > Deployed the source code on heroku.com > Author of chapter III.
	Diego Gaboardi (Computer Science)	> Implemented the dynamic adaption of the user interface regarding the data from the database > Implemented the structure of the user interface > Author of chapter III.
	Lukas Kick (Business Information Systems)	> Designed (UI Mockups - appendix 5.3) and implemented the user interface of the FleetMe platform > Created an overview of the available data and a selection for the FleetMe service > Created and presented the prototype to an audience of selected BMW employees > Calculated a business case forecast > Author of chapters I., IV. and summary
	Muhammad Asad (Computer Science)	> Filmed and created the trailer/video for the final presentation > Wrote the majority of the blog entries > Author of chapter II.

Table 1: Project contribution of the team members
Source: own illustration

1.3 Foundations: Car as a Service Market & Fleet Management

According to a Roland Berger study, the Car as a Service market will undergo a constant growth in the next years. The proposed compound annual growth rate (CAGR) is indicated to reach a level of 3.3 percent in the years 2019 to 2025 (see Figure 1). Due to the estimated market volume of 53 billion in 2019, fleet management lies within an emerging market while offering

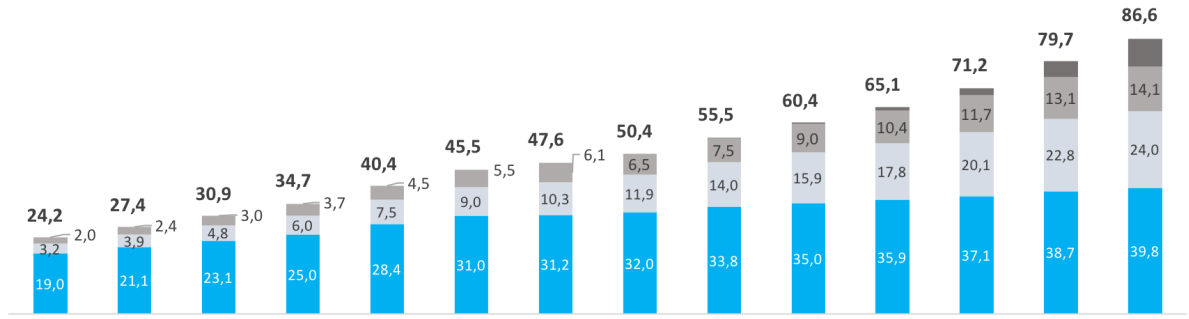


Figure 1: Car as a Service Market development in the years 2000 to 2025
Source: Roland Berger (2018)

great financial potentials (Berger, 2018, p. 50). In order to get a good understanding of the current structures in fleet management, the chapter at hand explains the characteristics of the two big players in the market and their different needs. This knowledge is mostly based on an expert interview that was conducted with a fleet manager from BMW.

On the first level, most companies within the Fleet Management market can be categorized into either a leasing company or a small/medium or large-sized company that operates a vehicle fleet. Whereas the leasing companies own the vehicles as they buy the cars directly from the car manufacturers, the fleet companies lease the vehicles from the leasing companies for a specific time period that normally lies within half a year and up to four years. Through highly restrictive contracts, the fleet companies take over the ownership of the vehicles during the specified period of the contract. Most of the latter include regulations on the maximum mileage that can be driven with a car as well as the condition of the vehicles after the end of the contract. Therefore, the companies are urged to comply to the contractual terms in order to keep costs of the company car pool to a minimum. Major cost factors of company fleets include vehicle depreciation, fuel and maintenance costs, but also opportunity costs due to low utilized vehicles. An illustration of the players and most cost factors is given in the following Figure 2.

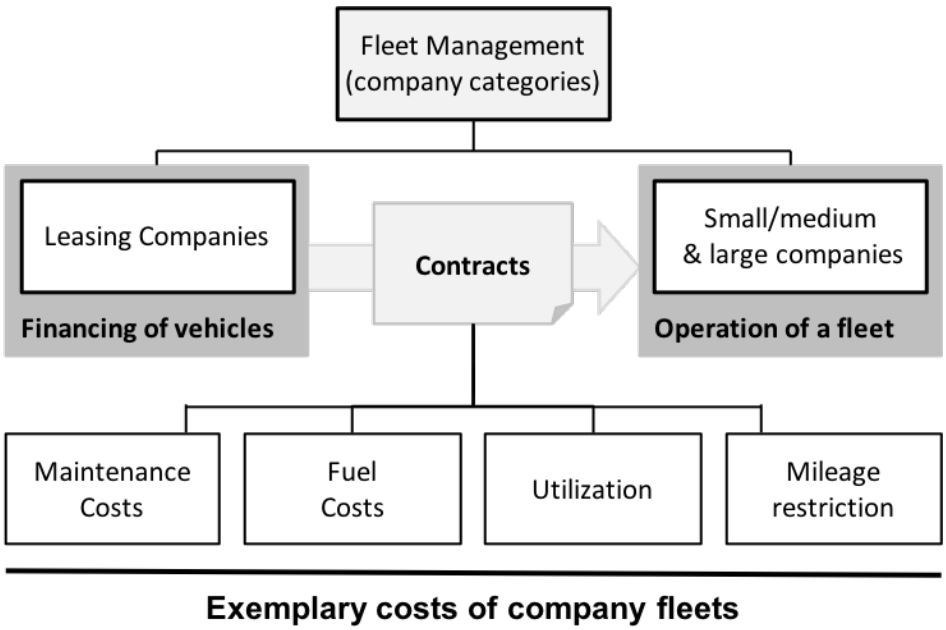


Figure 2: Overview of Fleet Management
Source: own illustration

Having those cost factors in mind, we think that costs can still be decreased and that the analysis of the driving behaviour of employees as well as a simplified vehicle maintenance process can help fleet managers in reducing overall company vehicle costs.

2 Utilizing the Design Thinking approach in the process

2.1 The general Design Thinking approach

Identify a problem, design possible solution(s), test solution, produce, and ultimately sell it - sounds simple enough. However, that is not so in perhaps 95% of the cases. The process of introducing a new product or service involves countless iterations of above mentioned steps, step-forwards, tracebacks, failures and reinventing over the whole idea to final product/service cycle. And even that does not necessarily mean that the final solution would be accepted by the end customer in its entirety.

At the end of the day, it is all about the customer. It does not matter if the newest and the most innovative of technologies was used to develop the solution, if it doesn't adapt to the customer's way of life and isn't relatable to them, the product/service is not going to sell. Which is why the process of product/service development should take a customer-centric approach to help in the finalization of a solution.

The approach we were introduced to over this practical course was to not only keep the customer in mind while identifying the problems we wanted to address, but also over the development, prototyping and ultimately testing phases. And then start over from the relevant step as necessary.

The following Figure 3 summarises the design thinking cycle taught over the course of the practicum:

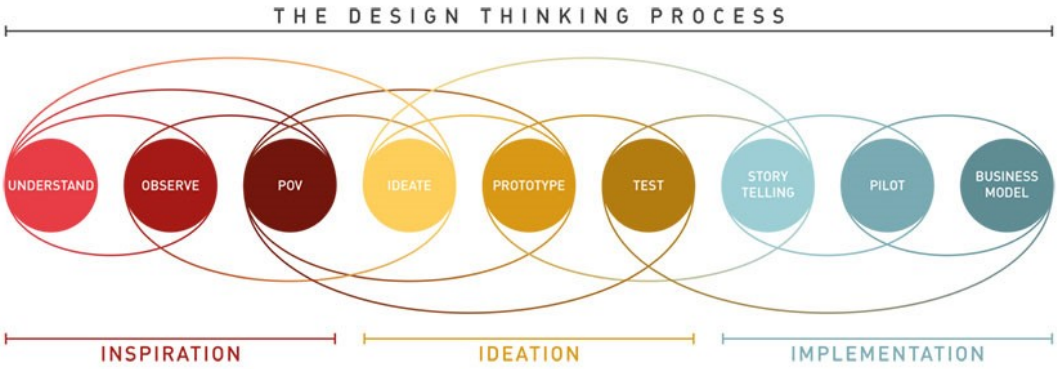


Figure 3: The design thinking process
Source: Design Thinking: A Quick Overview (2018)

It is important to emphasize again that all the above depicted steps were to be carried out with the customer in mind and involved. This process was introduced to the class with the help of small sample projects, such as designing a transport vehicle for tourists within cities by working in pairs, where one student acted as the tourist, and the other as the designer. The designer should understand the "Who? Why? Where? What? How? When?" of the end user and synthesize a point of view.

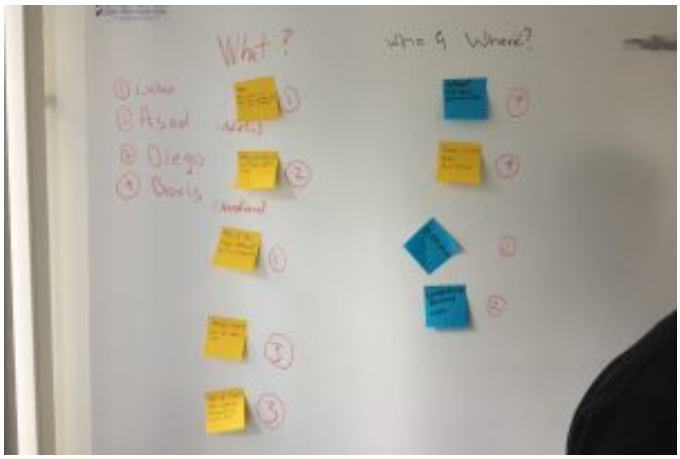


Figure 4: Brainstorming the next steps
Source: own illustration

Once that's done, the next step is to come up with ideas for solutions that can address the problem keeping the above users' point of view in mind. The more ideas for solutions, the better it is. Then the most feasible of these ideas was to be chosen and developed. Once a prototype of the solution was developed, it was tested with the respective tourist and feedback recorded. Although this exercise was an over-simplification of the actual final design process, it helped the students gain an important insight into how to develop a product or service while keeping the customer's point-of-view, needs and thought process in mind, and actively engage them as well to be able to improve the solution in next iterations.

In essence, the fundamental principles behind design thinking can be summarised as follows:

> Design Thinking starts with empathy, a deep human focus, in order to gain insights which may reveal new and unexplored ways of seeing, and courses of action to follow in bringing about preferred situations for business and society.
> It involves reframing the perceived problem or challenge at hand, and gaining perspectives, which allow a more holistic look at the path towards these preferred situations.
> It encourages collaborative, multi-disciplinary teamwork to leverage the skills, personalities and thinking styles of many in order to solve multifaceted problems.
> It initially employs divergent styles of thinking to explore as many possibilities, deferring judgment and creating an open ideations space to allow for the maximum number of ideas and points of view to surface.
> It later employs convergent styles of thinking to isolate potential solution streams, combining and refining insights and more mature ideas, which pave a path forward.
> It engages in early exploration of selected ideas, rapidly modelling potential solutions to encourage learning while doing, and allow for gaining additional insight into the viability of solutions before too much time or money has been spent.
> Tests the prototypes which survive the processes further to remove any potential issues.
> Iterates through the various stages, revisiting empathetic frames of mind and then redefining the challenge as new knowledge and insight is gained along the way.
> It starts off chaotic and cloudy steamrolling towards points of clarity until a desirable, feasible and viable solution emerges.

2.2 Our individual Design Thinking approach

The aim of this practical course was to develop a product or service that makes use of BMW's CarData service to solve a problem or address a need. The first step required for this was to understand the people that fall under this design space (i.e., existing and/or potential users of BMW ConnectedDrive). This was done using secondary (using existing statistics and studies) and primary research (interviews etc.) in order to create profiles of potential customers and user groups (in the form of personas to establish empathy and understanding). Since we wanted to develop a B2B service, our team created two personas using the above-mentioned research methods and then developed 2 user personas on a life-sized chart to better tell the point of view story. The personas we developed were both mid-level employees at a consulting firm with different jobs, personal characteristics, need to travel and behaviour towards technology.

The next step was to brainstorm potential business opportunities in the form of products and services that the identified user groups would be able to benefit from. All groups were encouraged to go for as many ideas as possible, work off of ideas that others have brainstormed, get feedback and provide feedback to others on their ideas. This was done with the help of filtering the brainstormed ideas and narrowing them down to three ideas in the form of detailed posters that would help entertain the needs of the personas. The posters developed by our team revolved around a company fleet car pricing dashboard, custom car insurance premium calculator based on driving style of the driver and electric car battery optimizing system. Apart from these detailed posters that directly addressed the needs of our personas, each team was required to develop 2 idea napkins for other ideas. All of these were presented in class and feedback was gathered about the feasibility, practicality and potential of the ideas.

After many iterations of feedback and redesign of our idea and solutions, the ideas were refined to two finalists (dashboard and electrification) and ultimately merged into one single solution as a fleet management system that could optimise vehicle upkeep, usage and maintenance. And this way, the concept for FleetMe Fleet Management System was born.

The next step was to develop and present a physical prototype for this solution to visually depict how or solution aims to relieve the target user groups. The positives, negatives, possible improvements and suggestions for improvement about our idea were collected from the rest of the class and the instructors to help refine the solution. All the users had actually physically used the prototype and with the help of a user story helped them get into the shoes of the target group. After this the prototype was further refined and enhanced based on the feedback synthesised and a pitch for the idea was developed. Our pitch was based on the assumption that all companies aim to improve their overall profits. And the two possible ways to do this is to increase revenue or decrease costs, FleetMe aimed to help companies increase profits using the latter method.

In order to make a product or service viable, it is essential that it not only solves a problem but run it as a business, bring in revenues and ultimately profits. How our ideas could achieve this was done by developing suitable business models. Our team made use of a business model canvas and value proposition canvas to depict the customer, production activities, resources and how FleetMe could earn money. As a result of this we were able to document the exact customer segment, what their needs are, what we would offer, what resources are product would use, the costs and how we plan to bring in the money using a subscription model for revenue.

Ultimately after further several iterations of prototyping, gathering feedback, and redesigning we set up FleetMe as a web application using live data from BMW CarData.

3 Our Solution: FleetMe (Prototype)

3.1 Technologies

3.1.1 Server implementation: Node.js & express framework

Node.js fits our application because it handles well with high volume of short messages requiring low latency. Especially for real time applications node.js backend is best fit. In our case, we need to send "get" request to the BMW server for every car in our fleet in every five minutes or sometimes even more frequent. Therefore, node.js can easily handle this load.

Other advantage of node.js is that in collaboration with express module it provides us a ready to use API structure. This means that from any kind of frontend system our backend will be able to provide necessary data via the API. For instance, from a mobile platform or web service, we can send get request to our API created by the node.js backend. This request can take the form of a url such as "www.fleetme.com/api/trips/all", this url fetches every trip saved in the database.

The other advantage of having an API is that the development of front-end and the back-end are totally separated. This becomes really important in agile development processes. For example, front-end development does not need to wait for the back-end development to provide the necessary structure. Development process can work vertically and during development team can produce prototypes faster than the traditional use cases. Since in Advance Mobility Course, we needed to present our prototypes every two weeks, it is beneficial to use a system that enables us to proceed in vertical development.

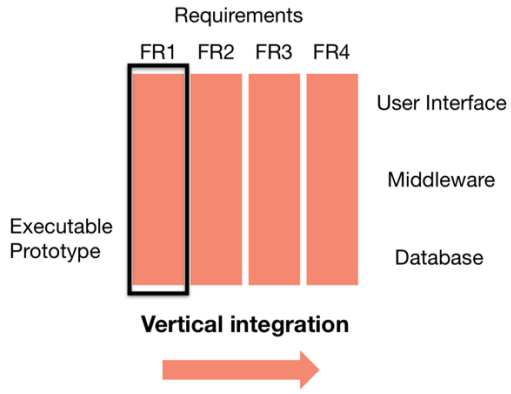


Figure 6: Vertical integration
Source: own illustration

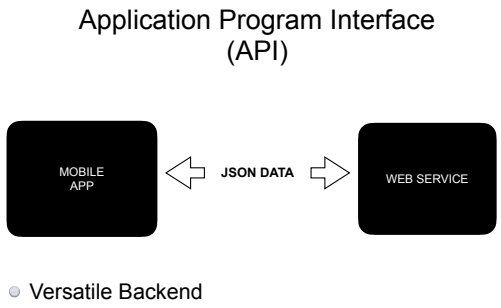


Figure 5: Application Programming Interface (API)
Source: own illustration

Express is a framework built on top of node.js. Express framework helped us to build server-side web application. Express provides easy to use tools for routing and helps to organize our JavaScript code into a MVC (model-view-controller) application.

In short express is a framework that helps to structure node.js into REST API.

3.1.2 Source code distribution: Github (Git)

GitHub is a web-based hosting platform for version control using “Git”. The choice of GitHub was based on the popularity of the service and every team member has their own GitHub account. Therefore, we could easily get started with GitHub.

The choice of version control system called “Git” is more complex. Git is a distributed version control system. In distributed version control system every developer has their own local repository with full history of commits.

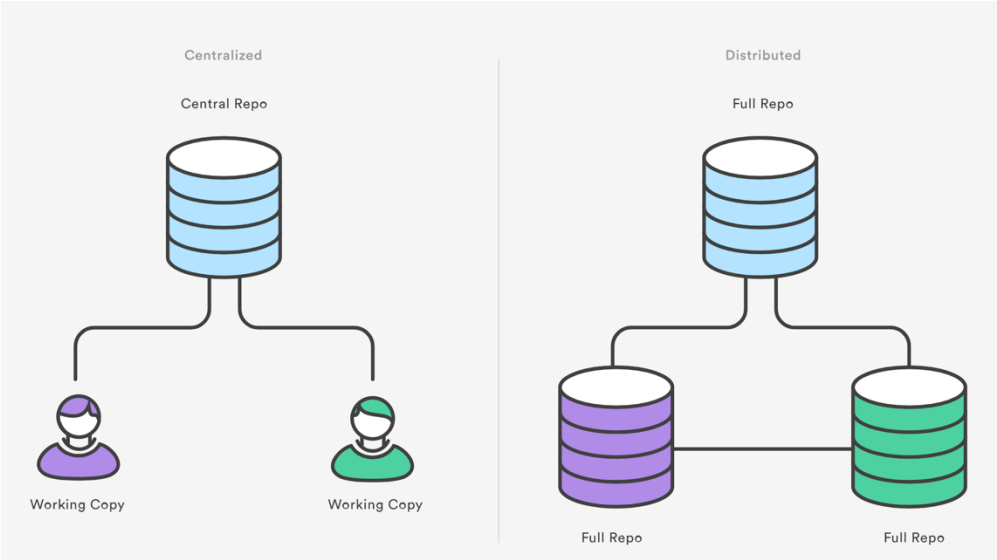


Figure 7: Centralized vs. distributed version control center
Source: own illustration

This structure provides developers a more durable and faster version control to use. On the other hand, centralized version control systems are more vulnerable to breakdowns. For example, if someone breaks the production branch other developers cannot commit their changes until the problem is fixed. This kind of breakdowns are obviously not possible in distributed version control system.

3.1.3 Database technology: MongoDB & Goose

MongoDB is a NoSQL document database. NoSQL databases are non-relational, meaning that they provide transactions in a way other than tabular relational methods which is the traditional way(MySql).
One of the advantage of MongoDB is that it provides expressive query syntax. It is more intuitive than SQL like query syntax. Hence it is easier to learn.
MongoDB is faster than traditional relational databases because it stores workable data at the RAM instead of Hard-Drive. Moreover, MongoDB is documented in developer friendly way, there are many free online courses and examples to get started easily. We have enrolled to this free course “*university.mongodb.com/courses/M101JS/about*”. In this course they teach developers know hows related to MongoDB to use for Node.js.

Mongoose is an abstraction module over pure MongoDB. Main advantage it provides is, it's schema structure on top of MongoDB's own NoSQL way. This schema structure resembles the SQL type of structure where most developers have experience with.

```
const mongoose = require('mongoose');
const Schema = mongoose.Schema

//Car Schema
const CarSchema = mongoose.Schema({
    vin:{
        type: String,
        required: true,
        unique: true
    },
    model: {
        type: String,
        required: false
    },
    image: {
        type: String,
        required: false
    },
    trips: [{
        type: Schema.Types.ObjectId,
        ref: 'bmwdatas'
    }]
})


CarSchema.index({vin: 1}, {unique: true});

const Car = module.exports = mongoose.model('Car', CarSchema);

// Finf6 Cars
module.exports.getCarsData = (callback, limit) => {
    Car.find(callback).limit(limit);
}
```

Figure 8: Example schema structure of Mongoose
Source: own illustration

Schema structure not only make the structure look more like SQL way, it also provides built-in validation and population methods. Those features are excessively used in our application. In Figure 8 the car-schema is developed with built-in mongoose schema structure in our application. In Figure 9 populate method of the mongoose is used to receive all trips of a particular employee.

```
//get trips of a user by id
router.get('/:id/trips', (req,res)=>{
    // console.log(req.params)
    try{
    User.findOne(req.params).populate('trips').exec((err,result)=>{
            if(err) return console.log(err)
            try{
            res.send(result.trips)
            }
            catch(e){console.log("No user found!!")}
    })
    }
    catch(e){
        console.log("No user found!!")
    }
})
```

Figure 9: Example of populate method of Mongoose
Source: own illustration

3.1.4 Server deployment: Heroku

Heroku is a hosting service platform. Heroku handles infrastructure changes and operations for developer. Developers only need to deploy their git repository of the application to heroku by "git push" command. They provide easy to use system with many free extra plugins such as mLab(MongoDb cloud database).

Heroku fits seamlessly to our application. It supports MongoDB and node.js, it provides easy to use interface, it allows free usage of their dyno and it uses “Git” version control system as means of deploying. Since in our application we need to provide rapid prototypes, it is better for us to use a “platform as a service” like Heroku to handle everything in the background for us.

In the future, if our priority switches to less latency, we would move to “infrastructure as a service” like AWS (Amazon Web Services) to gain more speed during transactions, in our application.

3.1.5 Database deployment: mLab

MLab is cloud database service specifically for MongoDB databases. We have used mLab cloud database for development and production purposes.

Production purpose use is the usual use case. Moreover, in our case we needed to ensure every group member has the same data while they are developing locally. Thus, we integrated mLab also in to our development machines.

There are couple of other cloud database providers such as MongoDB atlas. MongoDB atlas is preferable over mLab, if our main intention was security and backup & recovery option.

In our case basic security settings and backup & recovery options are suffice. Since, we have developed for prototyping purposes. However, in the future we should migrate to MongoDB atlas which is more expensive but more secure and flexible. Figure 10 depicts the easy interface to interact with mLab cloud database.

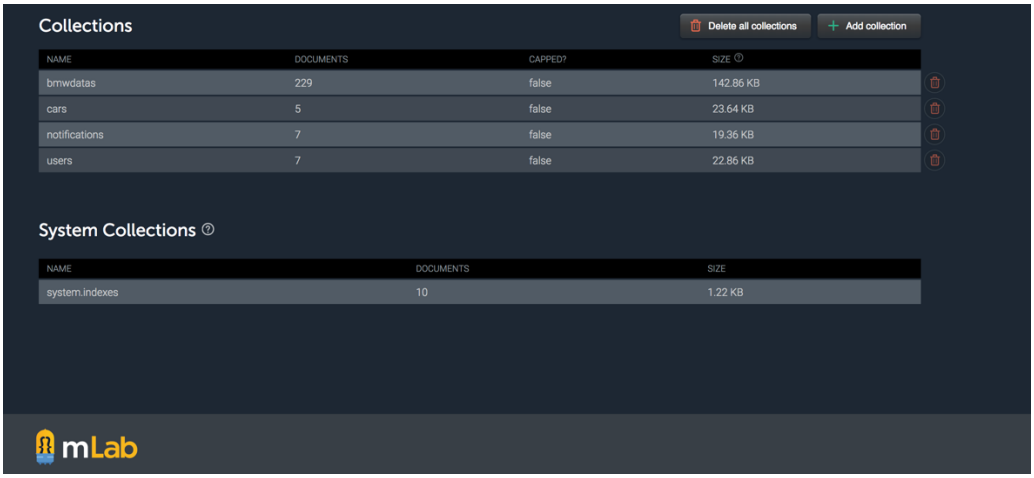


Figure 10: User Interface of mLab
Source: own illustration

3.1.6 User Interface technology: ReactJS & CSS

In order to make our application dynamic we decided to use **ReactJS**, a JavaScript library maintained by Facebook, Instagram and a community of individual developers and corporations.

The main advantage of this solution is the fact that we were encouraged to create reusable User Interface components called "*components*". In this way our application was readable easily maintainable and at the same time extremely flexible.

React component is a single object that not only outputs HTML like a traditional template would, but also includes all the code needed to control that output. React's data flow is unidirectional: data can only go from parent components to their children through "props" which is an abbreviation for "properties". When a component needs to react to data that doesn't come from a parent component we use "state" that constitute its temporary store.

Another reason for using ReactJS is the fact that it uses Virtual DOM which increases the performance of highly loaded applications and eliminates the potential inconvenience and improves user experience.

All the components are stored in the "TUM_Mobility_Services/src/app/component" folder and are written in upper Camel Case. They consist of the definition of a class with the same name of the component file in which there is a constructor and a render method.

For the presentation of our document we decided to use Cascading Style Sheets (CSS) so that we were able to enable the separation of presentation and content.

All components use **Bootstrap,** a free and open-source front-end library for designing websites and web applications which contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions.

At the same time, we decided to further personalize the style of our components writing our own CSS files for almost all the components.

These files are stored inside the folder "TUM_Mobility_Services/src/app/css" and have the same name, in lower Camel Case, of the component in which they are imported.

3.1.7 Additional frameworks & services

These are the other services used in order to create our application:

- Google Map API with single Marker to show the last position of the car during a trip;
- Google Map API with multiple Marker to show the current position of all the cars;
- React-rangeslider to create the sliders in the configuration panel;
- React react-tooltip to create the info button;
- font-awesome to obtain several icons.

3.2 Features of our prototype

The aim of this section is to present an overview of the most important features of our prototype. All these functionalities are available through a web browser and they can be seen at this

link: http://bemostwanted.herokuapp.com/. We recommend using Google Chrome browser to have the best experience with our application.

The main goal of our application is to gather and analyse data from the company cars in order to optimize their utilization and reduce costs. We can distinguish three main features:

3.2.1 Vehicle Management

In Vehicle Management we analyse the utilization and the status of vehicles in order to obtain an optimized fleet size, to reduce delays and to simplify the maintenance. On the upper part of the page the current position of all cars can be seen (Figure 12). Clicking on the label of the specific car it is possible to analyse the details of the specific car (Figure 11).

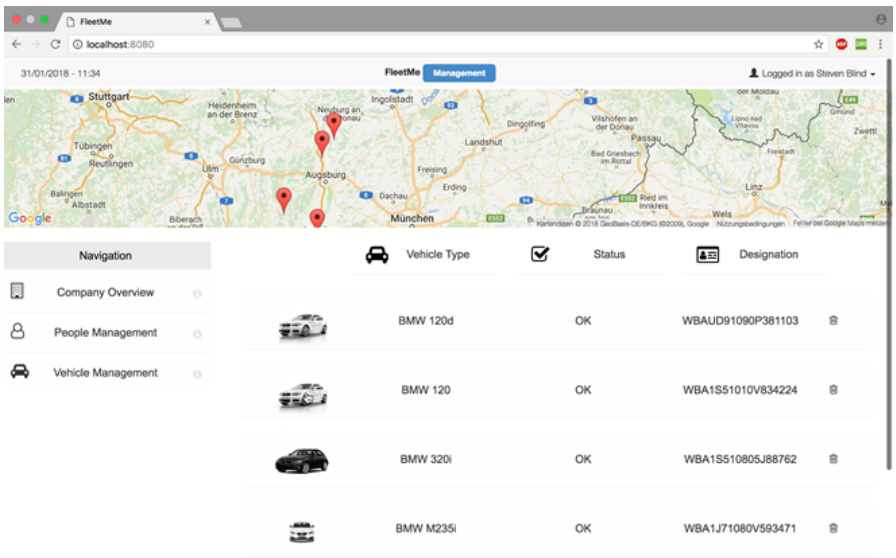


Figure 12: Vehicle management component
Source: own illustration

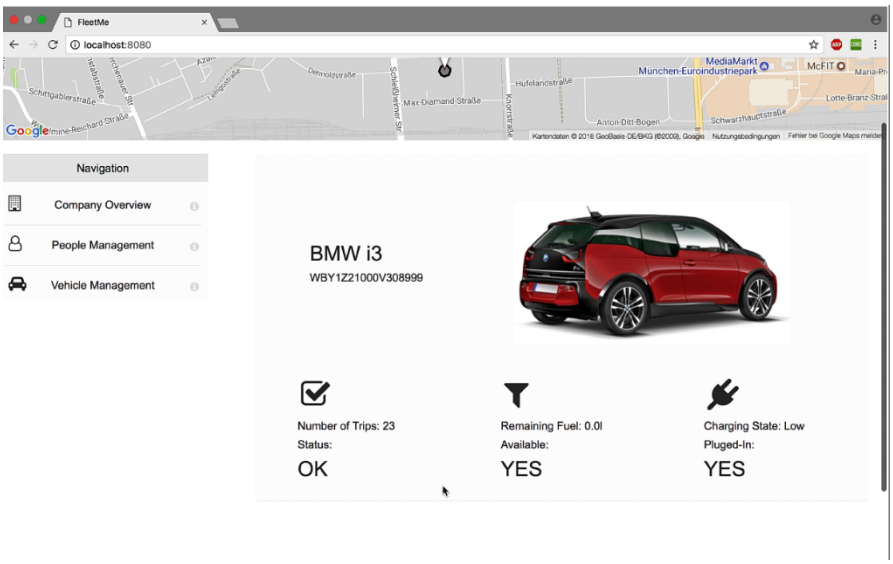


Figure 11: Overview of vehicle status
Source: own illustration

3.2.2 Employee Management

In Employee Management we analyse the behaviour of employees in order to give them recommendations and so to reduce costs due to bad driving style. Employee can be sorted in increasing or decreasing order in function of the overall rating that takes into account the driving behaviour of all trips. Clicking on the employee label (Figure 13), the list of all trips can be seen (Figure 14) and it is also possible to analyse the details of the specific trip (Figure 15).

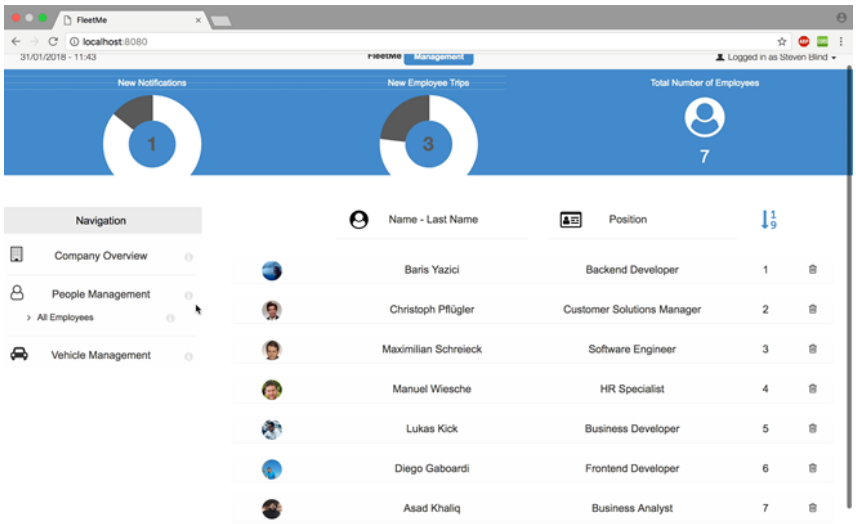


Figure 13: Employee management component
Source: own illustration

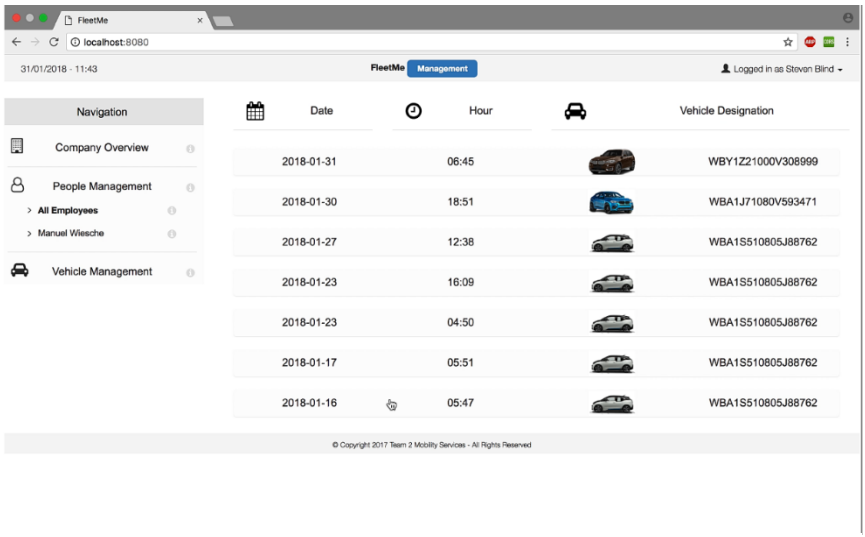


Figure 14: Overview of all trips of an employee
Source: own illustration

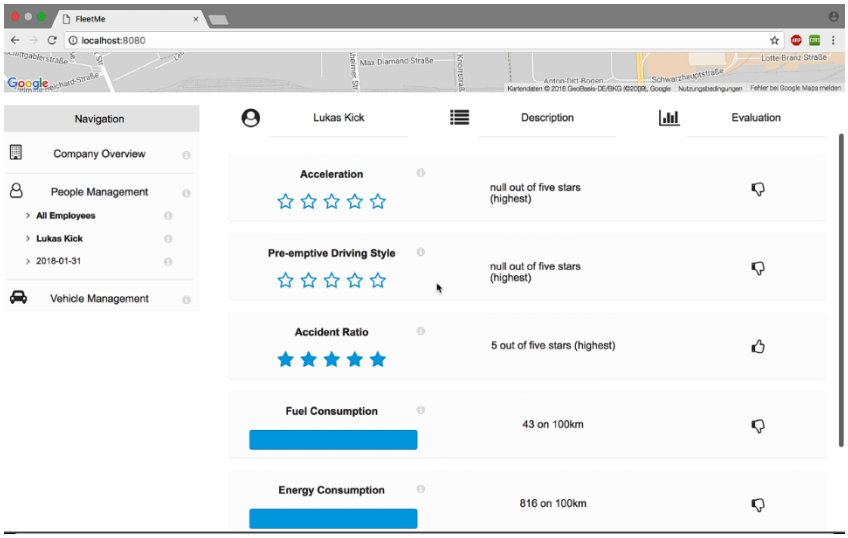


Figure 15: Driving behaviour analysis of a specific trip
Source: own illustration

3.2.3 Notification Centre

In Notification Centre we show to the fleet manager the most relevant problems in an intuitive way. The notifications can be filtered so that it is possible to find only those important for the fleet managers. Clicking on the notification label, it is possible to analyse the details of the problematic trip or vehicle. On the configuration panel it is possible to customize the thresholds and so the value that are used to trigger a new notification.

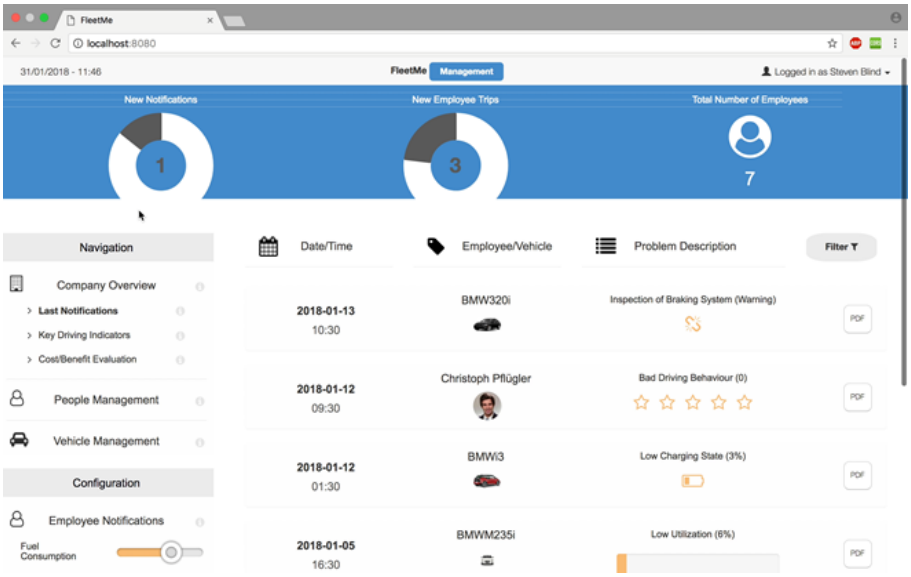


Figure 16: Notification component
Source: own illustration

3.3 Development road map

Informing the team members about use case and the structure of the back-end system is crucial part of the communication in the development part of the team. Team members except the back-end engineer, might not need to know how exactly things work around in the back-end.

Nevertheless, they receive basic information related to the general process followed in the back-end. Hence, they become more conscious about the technologies and the structure being followed. Figure 17drawing depicts the development road map. The linkage between the front-end and the back-end is critical for the developers because they structure their code based on that connection.

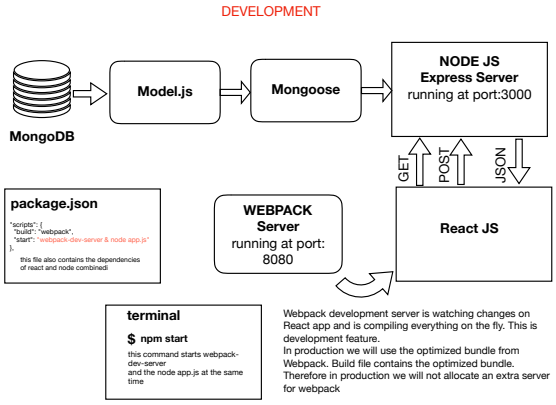


Figure 17: Development roadmap
Source: own illustration

3.4 Production road map

Production road map is slightly different than the development road map. Because, in Heroku (PaaS- Host Service) there can only be one server running. Therefore, webpack server is eliminated. Then, React.js code is built and rendered from app.js file which is the main file. Second difference is that in production mLab cloud database system is used instead of local database.

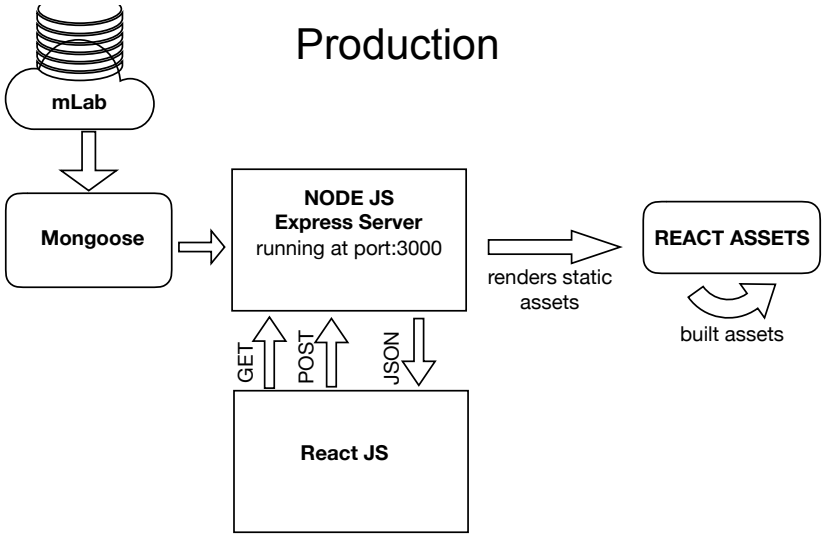


Figure 18: Production road map
Source: own illustration

3.5 User Flow

When a fleet manager logs in to FleetMe, the first thing they render is static react.js assests in front-end. After they log in they receive a token which is checked by subsequent steps to ensure user is authenticated. In Figure 19 fleet manager send a get request to the api/vinBmw endpoint of the web service. Here it is important to acknowledge that token is sent together with the get request to ensure the request is coming from an authenticated user.

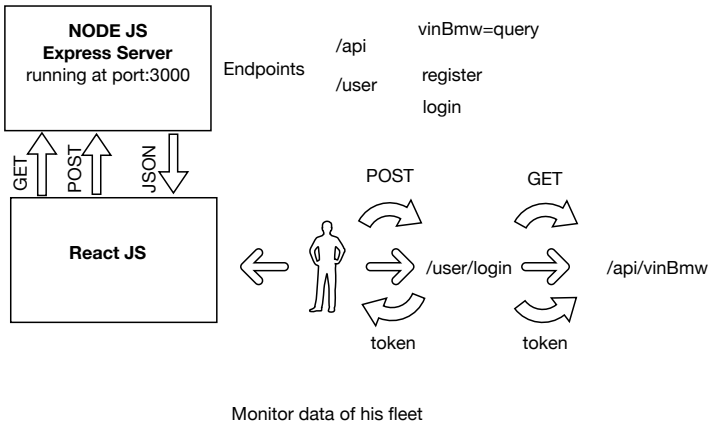


Figure 19: User flow of the FleetMe web service
Source: own illustration

3.6 Experiences with the BMW CarData API

BMW CarData API introduces an easy to use and integrate interface. We fetch data from BMW server via the "request" module on node.js.

Most of the problem we have, are related to inconsistency on the system.

- First problem is that the use case of car sending the data to the server, it differs heavily from model to model even in the same model it differs with year it was produced. For some cars, it sends data after it is locked from outside. For others it sends data when the mobile application is opened.
- Second problem is; when some features are not available in a certain model it gives "null" result for that field. However, in electric cars (i3 in our case) remaining fuel feature is always "0" instead of being "null". I3 is not hybrid, therefore fuel option is not available on i3. Thus, "null" value should be sent.
- Threshold for the acceleration and braking stars are too high. We are always receiving 0 stars from each car in the system. Driver needs to be highly delicate to score higher than 0.
- For some applications it would be better to be able receive data while car is running and on the road.

4 Business & Financial Evaluation of FleetMe

As described in chapter 1.3, the Car as a Service market offers great investment opportunities, in particular for new ventures. Our service aims at small-, medium- and large-sized companies that operate a company fleet for their employees. From a business perspective we first took a look at existing companies in the latter market to identify possible problems as well as entry points for a new service.

An overview of competitors in the market is given in the section 4.1, whereas chapter 4.2 introduces the final business model aligned to our service. On that basis, we calculated a complete business case for estimating the revenue potential and to proof that our business model is reliable (chapter 4.3). Apart from the BMW CarData platform, we added a forecast describing other business opportunities which we also can take advantage of with our solution (chapter 4.4).

4.1 Market and Competitor Overview

Currently there are many software services existing for managing company fleets. Most of them address the most cost-intensive cost factors such as mileage indication of the leased vehicles as well as a summary of the current vehicle status. While all of the latter require a connectivity box that has to be installed within the car, our service is free of hardware installations and at the same time offers more detailed vehicle information. An example is the assessment of the driving behaviour of employees and the thereby the closely connected recommendation system. Also, our service shows the most important problems directly in a smart notification centre that can be adjusted to the needs of the fleet manager. An aggregated overview of competitors in the market is shown in Table 2.

Company name	Overall assessment	Description of features
fleetster (2018)	◕	> Short summary of vehicle status > Contractual mileage indication > Vehicle reservation system > Price: 15 euros per vehicle per month
Fleetio (2018)	◔	> Tracking of vehicles > Route overview for drivers (last routes) > Trip reports of driving behaviour
AVRIOS (2018)	◔	> Focuses on the financial impacts of fleet management > Accounting system for fleet management
IMS (2018)	◔	> Tracking of vehicles > Contractual mileage indication > Arrive-on-time logistics services
Explanation (legend):		
●	offers all possible features	

◕	provides comprehensive fleet data
◑	provides most fleet data
◔	specializes in an area of fleet management

Table 2: Major competitors offering fleet management software solutions
Source: company websites

4.2 Final Business Model

The underlying framework of our business model is the business model canvas which comprises nine segments: key partners, key activities, key resources, value propositions, customer relationships, channels, customer segments, cost structure and revenue streams. Due to a further focus on improving environmental implications of vehicle fleets, the two dimensions, social and environmental costs and social and environmental benefits are added. Each of the elements is detailed in Figure 20.

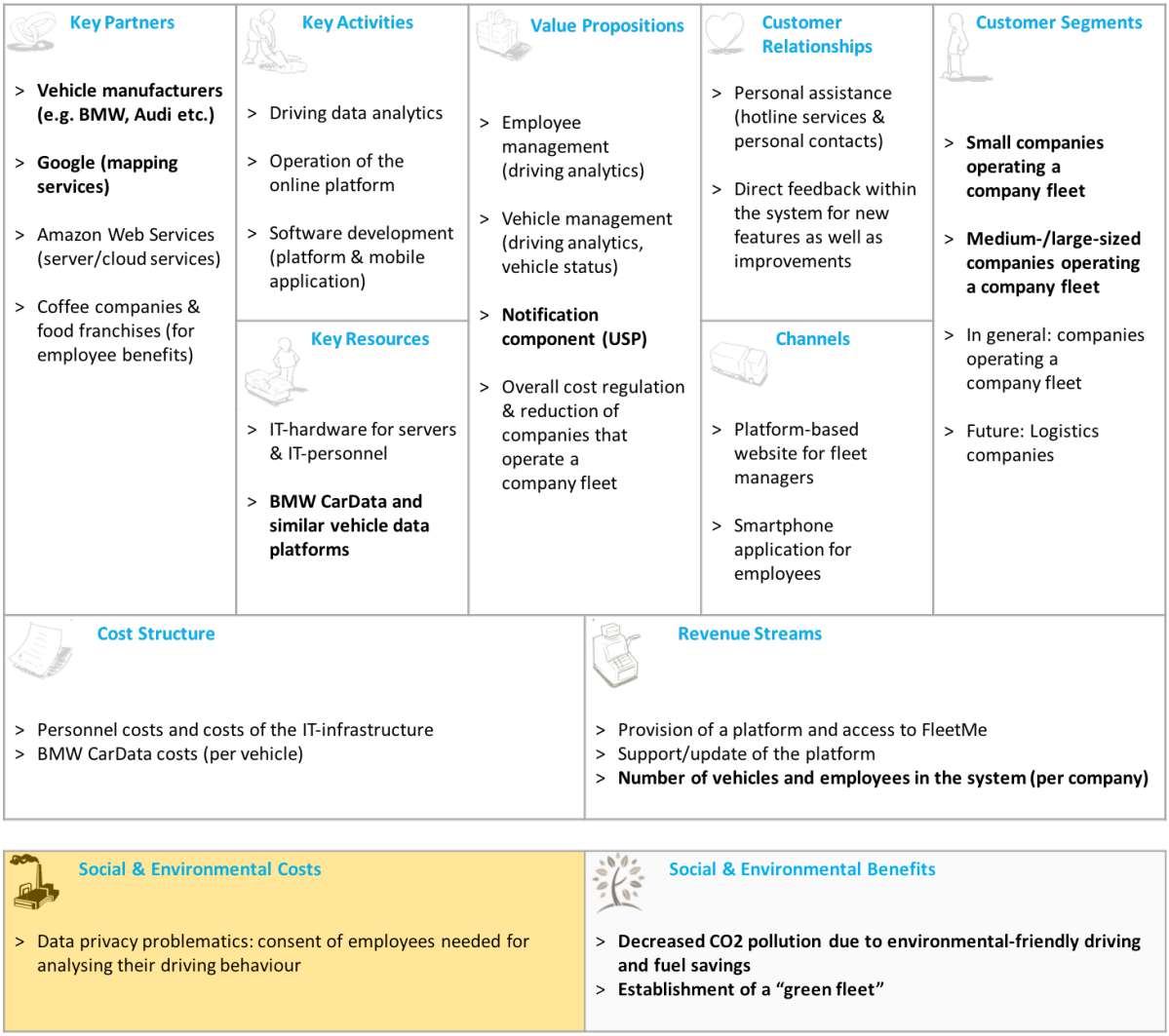


Figure 20: The FleetMe business model aligned to the Extended Business Model Canvas
Source: own illustration

4.3 Complete Business Case Calculation

Based on the business model that was introduced in chapter 4.2, we calculated a complete financial business case to plan the revenues in the years 2019 to 2026. This helped us also on providing a fair pricing strategy to our customers. In early 2022 we expect to reach the break-even-point.

Regarding our revenue structure, we divided our potential customers into two groups: small companies and medium-/large-sized companies. For the first group, we identified on average of 25 vehicles and 500 employees in the system, whereas for medium-/large-sized companies we calculated 400 vehicles and 20,000 employees. In the first year (2019) our focus lies on large-sized companies in order to have minor costs of marketing personnel. In the following years, we want to expand our customer base and significantly increase the numbers of customers. In late 2026 our customer size will include 1,040 companies, in which 40 companies are medium-/large-sized companies and 1,000 are identified as small companies. By increasing the number of customers, our revenue streams will proportionally scale. In 2019 we will make revenues of 1.5 billion euros, in 2022 it will increase to 12,3 billion and in 2026 more than 44 billion euros of revenues are expected.

Regarding our cost structure, personnel and IT-costs, in particular for hardware, implicate our highest expenses. Contrary to our expectations, costs for the BMW CarData platform will remain only a small cost item. Therefore, in 2019 we expect costs of 1,8 billion, in 2022 the expenses will increase to 8,5 billion and in 2026 costs amounting to 24,7 billion euros are estimated.

Altogether, our service will grow due to an increased customer base. In 2021, first positive earnings of 1,4 billion euros are expected. These will stay positive in the following years and increase significantly to 19,7 billion euros in 2025.

Regarding our pricing model, we also differentiate between small companies and medium-/larged-sized companies. For the both groups we price ten euros per vehicle per month. As we are also providing driver analytical components in our system, employees within the system are priced differently. In small company fleets, one employee in the system is charged by a monthly rate of five euors. Due to the fact that medium- and large-sized companies have significantly higher numbers of employees, the monthly rate is decreased to two euros (see Figure 21).

Figure 21: Pricing model of our service
Source: own illustration

The complete business case can be found in the appendix 5.1 as an attached business case excel sheet.

4.4 Future business opportunities

While in the previous business case calculation mainly companies that operate internal fleets such as consulting companies are addressed, companies in other sectors can also be taken into account. An example are transportation or packaging companies like DHL. Especially the latter companies operate very large vehicle fleets. Therefore, our solution could be individually adapted to their needs in order for them to decrease overall vehicle expenditures.

References

Atlassian (2018), https://www.atlassian.com/git/tutorials/why-git#git-for-developers (accessed 4 March, 2018)

AVRIOS. (2018). Fuhrparkmanagement Software 4.0. Retrieved from https://experience.avrios.com/fuhrparkmanagement-software?gclid=CjwKCAiAlfnUBRBQEiwAWpPA6e2WW2Hvs6cFcNb2B0B5hWhe5hXfzs97gR0U40hM-hsNLij5T5biEhoCnkcQAvD_BwE

Berger, R. (2018). Embracing the Car- as-a-Service model – The European leasing and fleet management market, 109. Retrieved from https://www.google.de/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwiJ28qh7tfZAhXImLQKHWQJBkEQFggyMAA&url=https%3A%2F%2Fwww.rolandberger.com%2Fpublications%2Fpublication_pdf%2Froland_berger_car_as_a_service_final.pdf&usg=AOvVaw2o9e1Yga4QbRE7HXBG1s8D

Bootstrap Website: https://getbootstrap.com/

Cascading Style Sheets: https://www.w3schools.com/css

Darel Lasrado, (2016), Database Zone, https://dzone.com/articles/mongodb-the-good-the-bad-and-the-ugly (accessed 4 March 2018)

Deloitte (2011): Deep diving for innovation, http://globalblogs.deloitte.com/deloitteperspectives/2011/10/deep-diving-for-innovation.html (accessed 11:45 pm, 4th March 2018).

Design Thinking: A Quick Overview (2018), https://www.interaction-design.org/literature/article/design-thinking-a-quick-overview (accessed 11:20 pm, 4th March 2018)

Fleetio. (2018). FLEET MANAGEMENT SOFTWARE: Track, analyze and improve your fleet operations with Fleetio. Retrieved from https://www.fleetio.com

fleetster. (2018). Software für Flottenmanagement, CarSharing und Vermietung: Cloud-Software für innovative Fahrzeugverwaltung Retrieved from https://www.fleetster.de

IMS. (2018). Safer. Smarter. Greener. - Leading connected car services and data insights provider to insurers, governments and enterprises. Retrieved from https://www.intellimec.com

Justyna Rachowicz (2017), https://www.netguru.co/blog/use-node-js-backend (accessed 4 March 2018).

Rachit Agarwal, **(2014)**, AlgoWorks, http://www.algoworks.com/blog/why-use-expressjs-over-nodejs-for-server-side-coding/ (accessed 4 March 2018)

ReactJs Website: https://reactjs.org/.

React: The Virtual DOM: https://www.codecademy.com/articles/react-virtual-dom.

Tayste, (2018), https://en.wikipedia.org/wiki/GitHub (accessed 4 March 2018)

Verischerungswirtschaftheute. (2017). Insurtech Friday kooperiert mit BMW. Retrieved from http://versicherungswirtschaft-heute.de/unternehmen-management/insurtech-friday-kooperiert-mit-bmw/

5 Appendix

5.1 Financial Business Case

Complete Business Case_F3.xlsx

Business Case - FleetMe System (2019 continuing)

Year	Ramp Up 2019	2020	2021	2022	2023	2024	2025	2026	
Revenues									
General Revenue Assumptions									
# Medium & Large corporations	3	6	12	20	25	30	35	40	
Average vehicle size	400	400	400	400	400	400	400	400	
Average number of employees	20000	10000	10000	10000	10000	10000	10000	10000	
Vehicle Costs in the system (one vehicle per month)	*10*	*10*	*10*	*10*	*10*	*10*	*10*	*10*	$
Employee Costs in the system (one user per month)	*2*	*2*	*2*	*2*	*2*	*2*	*2*	*2*	$
# Small orporations	0	40	100	200	300	500	700	1000	
Average vehicle size	25	25	25	25	25	25	25	25	
Average number of employees	500	500	500	500	500	500	500	500	
Vehicle Costs in the system (one vehicle per month)	*10*	*10*	*10*	*10*	*10*	*10*	*10*	*10*	$
Employee Costs in the system (one user per month)	*5*	*5*	*5*	*5*	*5*	*5*	*5*	*5*	$
Total Revenues	**1.584.000**	**3.048.000**	**6.756.000**	**12.360.000**	**17.100.000**	**25.140.000**	**33.180.000**	**44.520.000**	$
Medium & Large corporations	*1.584.000*	*1.728.000*	*3.456.000*	*5.760.000*	*7.200.000*	*8.640.000*	*10.080.000*	*11.520.000*	$
Vehicles	144.000	288.000	576.000	960.000	1.200.000	1.440.000	1.680.000	1.920.000	$
Employee's	1.440.000	1.440.000	2.880.000	4.800.000	6.000.000	7.200.000	8.400.000	9.600.000	$
Small corporations	*0*	*1.320.000*	*3.300.000*	*6.600.000*	*9.900.000*	*16.500.000*	*23.100.000*	*33.000.000*	$
Vehicles	0	120.000	300.000	600.000	900.000	1.500.000	2.100.000	3.000.000	$
Employee's	0	1.200.000	3.000.000	6.000.000	9.000.000	15.000.000	21.000.000	30.000.000	$
Expenses									
General Cost Assumptions									
Personal									
Fixed IT-Developers	5	10	15	20	30	30	30	30	
Fixed IT-Developers (wages)	100000	100000	100000	100000	100000	100000	100000	100000	$
Fixed Sales Personal	2	5	10	15	20	30	40	60	
Variable IT-Developers (per customer)	3	6	12	20	25	30	35	40	
Variable IT-Developers (wages)	80000	80000	80000	80000	80000	80000	80000	80000	$
Management personal	4	10	3	4	6	6	6	6	
IT Costs									
#Servers	18	116	272	520	750	1.180	1.610	2.240	
Average Server costs per server	5.000	5.000	5.000	5.000	5.000	5.000	5.000	5.000	
Variable CarData Costs (e.g. BMW) per vehicle	5	5	8	4	2	2	2	2	
Total Expenses	**1.856.000**	**4.601.000**	**5.507.200**	**8.586.560**	**12.026.472**	**15.598.766**	**19.172.720**	**24.753.664**	$
Fixed Costs									
Personal costs	1.500.000	3.500.000	3.100.000	4.300.000	6.200.000	7.200.000	8.200.000	10.200.000	$
IT-System costs (Software and Hardware)	20.000	24.000	28.800	34.560	41.472	49.766	59.720	71.664	$
Variable Costs									
Personal costs	240.000	480.000	960.000	1.600.000	2.000.000	2.400.000	2.800.000	3.200.000	$
IT-System costs (Software and Hardware)	90.000	580.000	1.360.000	2.600.000	3.750.000	5.900.000	8.050.000	11.200.000	$
CarData costs	6.000	17.000	58.400	52.000	35.000	49.000	63.000	82.000	$
Earnings									
	Ramp Up								
Prospective earnings	**-272.000**	**-1.553.000**	**1.248.800**	**3.773.440**	**5.073.528**	**9.541.234**	**14.007.280**	**19.766.336**	$

5.2 Interview guide

Interviewleitfaden/Diskussion

Agenda

I. Vorstellung Team „Be Most Wanted“ (Lukas Kick) (5min.)
II. Vorstellung Herr Svircev (BMW) (5 min.)
III. Allgemeine Fragen zum Thema Fleet Management (15 min.)
IV. Überblick über das Projekt: „FleetMe: Fleet Management System“ (30 min.)
   a. Projektdurchführung & Projektscope (5 min.)
   b. Diskussion: Aktueller Prototyp (20 min.)
   c. Zusammenfassung (10 min.)
V. Abschlusszitat (5 min.)

Fragenkatalog

Allgemein

1. In welche Segmente können Firmen (Corporates) hinsichtlich der Anzahl der Fahrzeuge und Mitarbeiter eingeteilt werden (z.B. Große Firma: 600 Fahrzeuge bei 60.000 Mitarbeitern)? Größe der Fahrzeugflotte?

2. Werden die Autos überwiegend geleast oder werden diese gekauft?

3. Wie viele Mitarbeiter werden zur Unterhalterung einer Flotte im Durchschnitt benötigt (z.B. ein Fleetmanager und zwei Fleet-Administratoren)?

4. Welche Tätigkeiten fallen in den Aufgabenbereich eines Fleetmanagers (primäre Tätigkeiten, Arbeitstag)?

5. Welche Kostenblöcke sind am größten (z.B. Maintenance)?

6. Was sind die größten Problemfelder im Bereich Flottenmanagement (z.B. Fahrverhalten)?

7. Welche Funktionalitäten bietet eine Standard-Fleetmanagement-Software und wird eine verwendet? Wenn ja, welche Verbesserungen sind erwünscht?

8. Welche Probleme, die in der Software nicht vefügbar sind, kennen Sie aus dem Bereich Fleet Management (z.B. Ladestatus/Tankstatus des Fahrzeugs muss manuell geprüft werden)?

9. Welche Features bzw. Funktionalitäten würden Sie sich von einem Fleet Management System wünschen (z.B. Notifikation zu geringem Tankstatus/Ladestatus)?

10. Welche Reporting müssen Fleet Manager ausführen (Ende des Jahres)?

11. An welchen Indikatoren werden Fleet Manager bemessen (z.B. Fuhrparkkosten, Benzinkosten pro Monat etc.)?

Prototyp: FleetMe

1. Wie würden Sie unseren Prototypen mit einem Satz beschreiben (z.B. sehr gelungen, ein neues spannendes Thema wird adressiert)?

2. Welche weiteren Funktionalitäten würden sich speziell im FleetMe Management System sehen?

3. Sehen Sie Funktionalitäten, welche im FleetMe System implementiert sind, und in der Regel nicht benötigt werden?

Weitere Prototyp-Evaluierung

- Design
- Übersichtlichkeit
- Benutzerfreundlichkeit
- Detaillierungsgrad

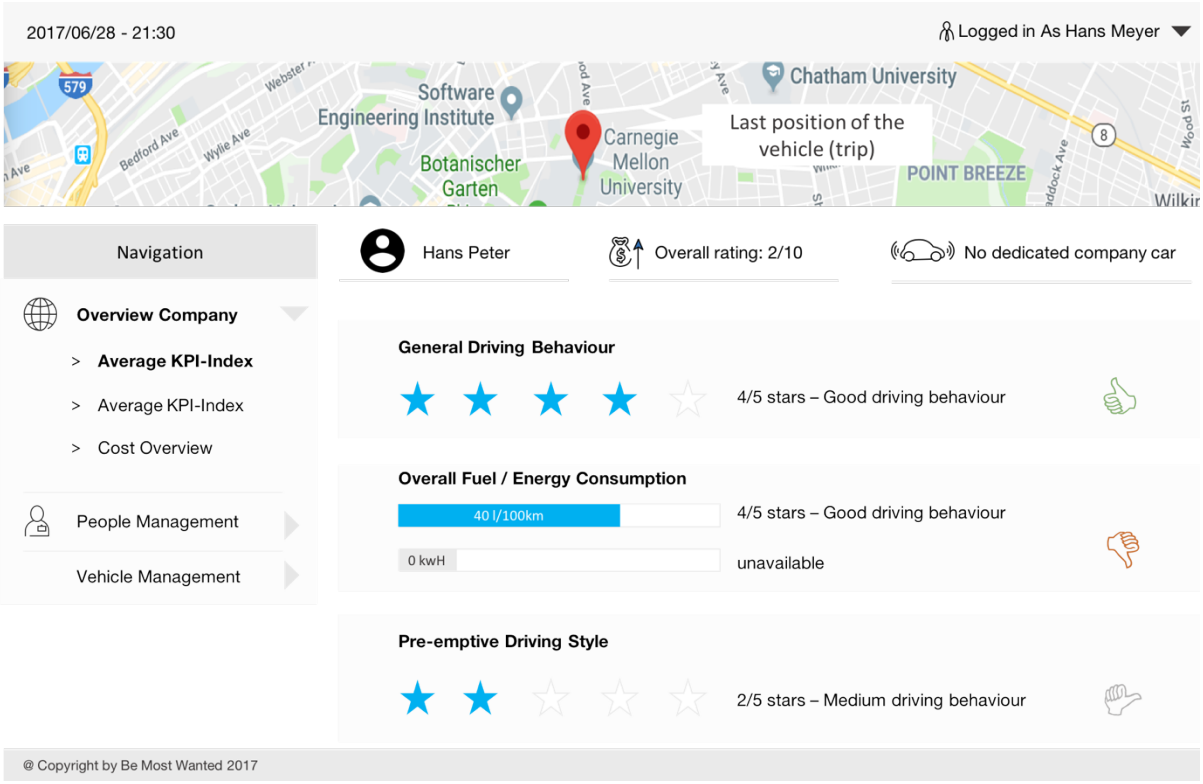
- Implementierte Funktionalitäten?
- Weitere Funktionen? – KDIs, E-Mail oder PDF?
- Incentivierungen für den Fleetmanager?

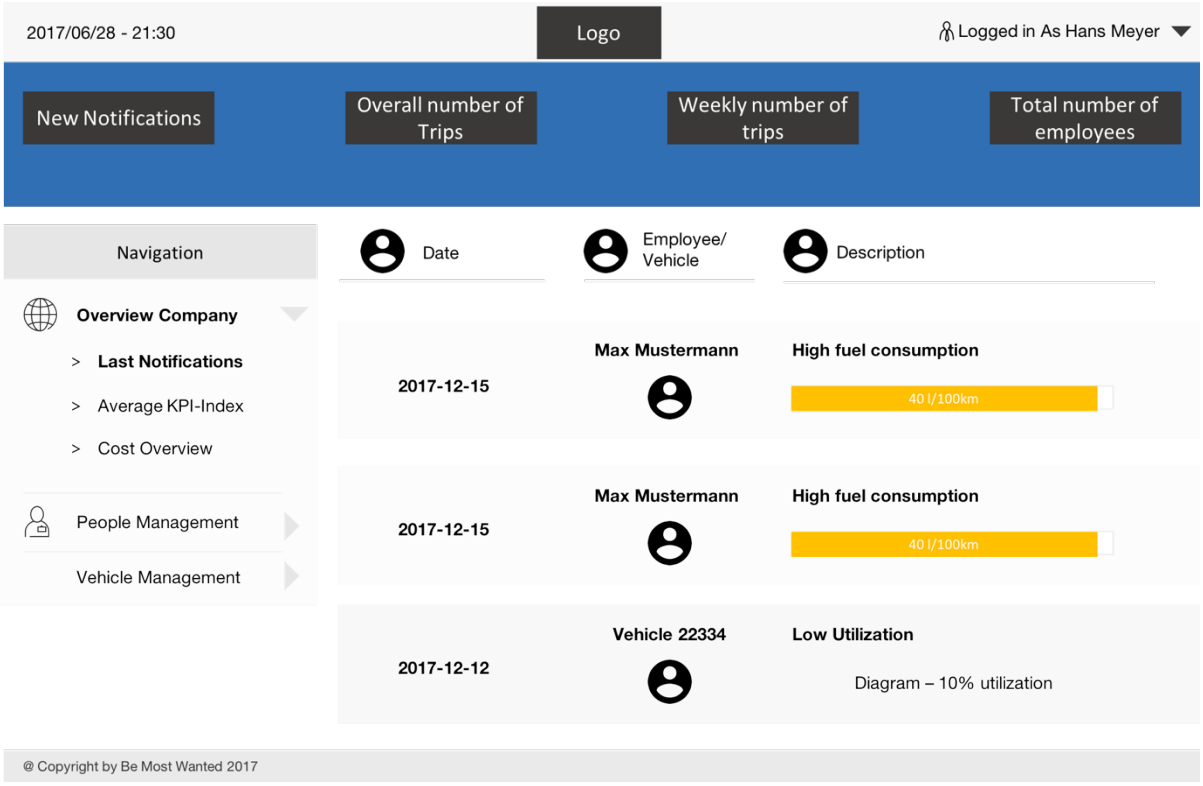
- Potenzial der Idee?

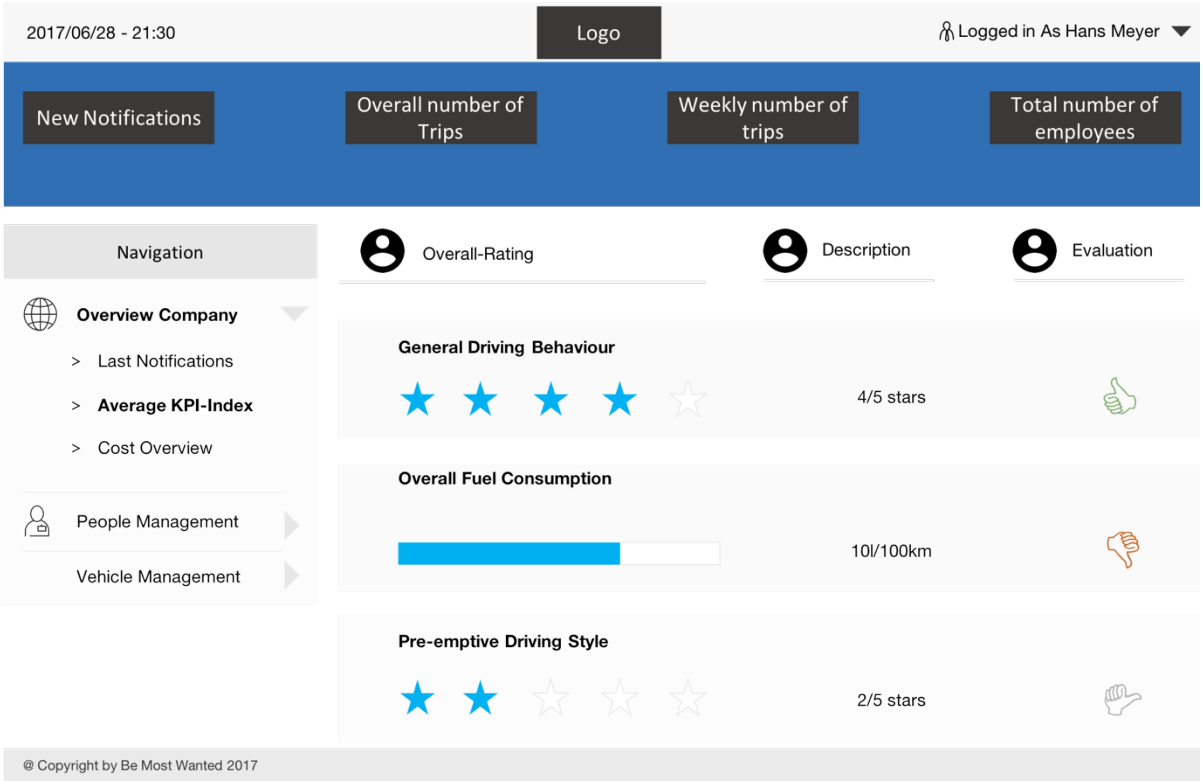
- Spezialisierung auf Elektrofahrzeuge?

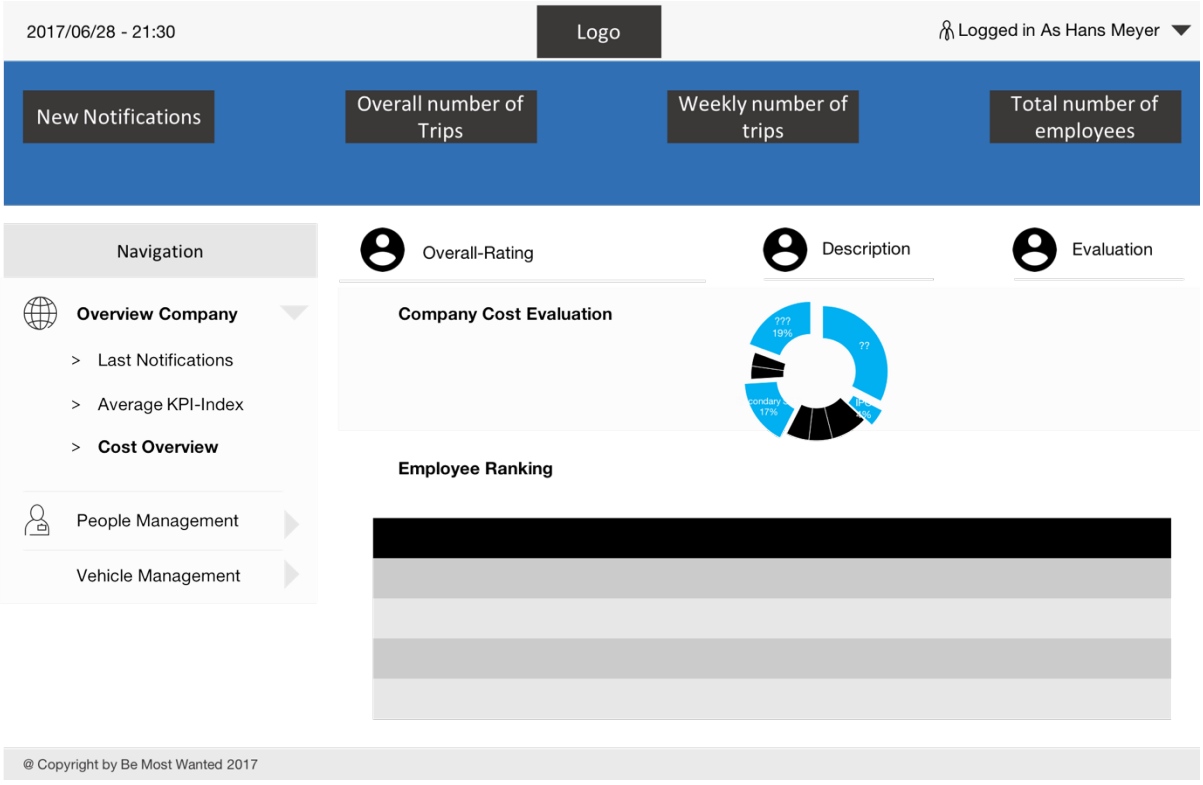
- **Bei Zeit:** Mobile Applikation für Fleet Manager oder Mitarbeiter?

5.3 User Interface Mockups






2017/06/28 - 21:30
Logo
Logged in As Hans Meyer
New Notifications
Overall number of Trips
Weekly number of trips
Total number of employees
Navigation
Overview Company
> Last Notifications
> Average KPI-Index
> Cost Overview
People Management
Vehicle Management
Overall-Rating
Description
Evaluation
General Driving Behaviour
4/5 stars
Overall Fuel Consumption
10l/100km
Pre-emptive Driving Style
2/5 stars
@ Copyright by Be Most Wanted 2017


2017/06/28 - 21:30
Logo
Logged in As Hans Meyer
New Notifications
Overall number of Trips
Weekly number of trips
Total number of employees
Navigation
Overview Company
> Last Notifications
> Average KPI-Index
> Cost Overview
People Management
Vehicle Management
Overall-Rating
Description
Evaluation
Company Cost Evaluation
Employee Ranking
@ Copyright by Be Most Wanted 2017