

UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Stanislav Belogrivov
Web Application for Business Parking Lot
Bachelor's Thesis (9 ECTS)

Supervisor: Siim Karus

Tartu 2018

Web Application for Business Parking Lot

Abstract:

Parking lots are an essential part of modern infrastructure. Due to the rapid development of cars as widely accessible transportation methods, the need for places, where vehicles could be stored when not needed, kept growing. Therefore, answering to this demand, the idea of parking lots came to be – a place that allows you to store your movable property for a certain amount of time without the fear that it will be moved elsewhere. However, due to the constantly growing amount of vehicles in use, especially in the production sector, because of the rapid technological advancement, the paper-based methods of handling parking lot workflow is becoming inefficient and obsolete. This problem become even clearer in case of trucks, where trailers are sorted. What is even more important is the uninterrupted work of such parking lots under conditions, when the process of leaving the parking lot requires utilizing limited in time services. In this particular case, national border crossing service. The aim of this thesis is to create a web-based alternative to the existing paper-based parking lot business processes. The thesis focuses on the digitalization, automatisisation and error-prevention of the said processes. In addition, modern business-processes demand the guaranteed servicing time (SLA). This thesis also provides a brief overview of how a parking lot operates currently.

Keywords:

Parking lot, web application, automatisisation, digitalization, SLA, service level agreement, accounting, business process research, business process description and mapping

CERCS: P170, Computer science, numerical analysis, systems, control

Veebirakenduste loomine äriparkla jaoks

Lühikokkuvõte:

Parkimiskohad on kaasaegse infrastruktuuri oluline osa. Autode kui laialdaselt kättesaadavate transpordimeetodi kiire arengu põhjal kasvab vajadus kohtade järele, kus sõidukeid saaks vajaduse korral hoida. Seepärast, vastavalt nõudlusele, tekkis parklate idee – need on ametlikud kohad, mis võimaldavad oma vallasvara hoida teatud aja jooksul ilma hirmuta, et see kaob või seda liigutakse kuhugi mujal. Et kasutusel oleva vallasvara arv suureneb kiire tehnoloogilise arengu tõttu, eriti tootmissektoris, muutuvad paberipõhised meetodid ebaefektiivseks ja aegunuks. Antud probleem muutub veelgi selgemaks veoautode puhul, kus haagised on sorteeritud. Veelgi olulisem on selliste parklate katkematu töö tingimustel, kui parklast lahkumise protsess nõuab piiratud ajaga teenindamist. Lisaks, niisugused meetodid suurendavad eksimuse riski inimfaktori tõttu. Käesoleva bakalaureuse töö eesmärk on luua veebipõhine alternatiiv olemasolevatele paberipõhistele parkimisprotsessidele. Töö keskendub mainitud protsesside digitaliseerimisele, automatiseerimisele ja vigade ennetamisele. Lisaks vajavad kaasaegsed äriprotsessid garanteeritud teenindusaega (SLA). Töö annab ka ülevaate sellest, kuidas äriparklad praegu töötavad.

Võtmesõnad:

Äriparkla, veebirakendus, automatiseerimine, digitaliseerimine, teenustaseme kokkulepe, raamatupidamine, äriprotsesside uurimine, äriprotsesside kirjeldus ja kaardistamine

CERCS: P170, Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria)

Table of Contents

1	Introduction	4
1.1	Problem Description	5
1.2	Research Questions	5
1.3	Preparation.....	5
2	Terms and Notations	7
3	Background	8
3.1	Research Methods	8
3.2	Research Results.....	8
	Restrictions and Requirements.....	10
3.3	Architecture and Structure.....	10
	APEX	11
4	Implementation	13
4.1	Solution Description.....	13
4.2	Specification of Business Processes.....	13
4.3	Database Development.....	15
4.4	Web Application Development	18
4.5	Business Logic Development.....	18
4.6	Version Tracking.....	19
4.7	Reports.....	20
4.8	Testing.....	21
5	Conclusion.....	22
6	Project Potential	23
6.1	Transfer all queries into the database	23
6.2	Automatic reports	23
6.3	Additional system integration.....	23
6.4	User role differentiation	23
6.5	Obtain extensive data for analysis	23
7	References	24
8	Appendix	25
I.	License.....	25

1 Introduction

The idea of preserving information in a written form for ease of use in the future is extremely old, dating back to 6th millennium BC [1, p. 20]. The understanding of the value of documentations and written forms became even clearer with the invention of the concept of trade. It became extremely important to solidify the agreement to avoid fraud or scam. Initially, it was possible to use the power of memory to manage trading operations, such as keeping historical records or financial accounts. However, with the expansion of such networks and, more importantly, the increase of the amount of trading operations, the use of memory became extremely inefficient. This forced the development of writing systems. The first documents regarding trade operations date back to fourth millennium BC. During those times, if the information was written on clay plates using cuneiform scripts.

With the advancement of technology, data storages switched to paper. The writing systems advanced to alphabets and numbers. With the discovery of machine printing technology, the templated data storing became possible, thus simplifying the client-service or service-service interaction [2, p. 19-28]. The use of templates allowed service providers to read and write necessary information way quicker as there was less information to write manually in.

Further research into the history shows, that each new discovery in this field has allowed simplifying and, as a result, speeding up the processing speed for services. Therefore, it becomes evident that the current development of digital technologies allow even more freedom in the field of automatisation and simplification of data transfer for service providers.

In the period of 1990-2000, computers and computational machines were used for relatively simple computational operations, which were quite often isolated from each other. After that, came the next revolutionary step in the automatisation practice - systematic use of information systems for business processes as whole objects.

However, the process of digitalisation simplifies the operations for end user only. The more advanced the technology becomes, the more standards are introduced and, thus, the more work must be performed in order to achieve the results. As of now, the digitalisation of any business process requires preliminary analysis and its description in a formal form.

As of now, the tendency to digitize business processes becomes popular even in relatively small business sphere. This work came from the fact that company Digiline OÜ approached author with this specific task: to analyze their business processes and to create an information system that could digitize the main process of their company. The system is supposed to be open and expandable as the solution provided for this paper does not intend to cover all processes of the company and will require further integration with the existing information systems in use, such as, for example, accounting information system, and, obviously functionality expansion, which will be described in the last part of the thesis.

This paper consists of two major parts:

- The Background chapter serves two purposes. Firstly, it describes the existing situation and, secondly, it provides an overview of the theoretical work of this paper - the research of business processes, data flow description, clarified the requirements for developed software: limitations, use cases, functional and non-functional requirements. Lastly, this part provides a brief summary of all the research results.
- The Practical part chapter, which follows the development process. It includes the description of the methods used, the problems encountered and their solutions.

1.1 Problem Description

The business in focus is a business parking lot owned by Digiline OÜ. The most crucial part of the business is processing the in- and outgoing transport. As stated earlier, the system relies on a paper-based approach, which Digiline OÜ now considers outdated and inefficient.

The paper-based approach, compared to digital one, has the following flaws:

- All tasks are performed manually by human operators, which affects process speed.
- Paper is less durable, which increases the risk of data loss.
- As paper documents can be divided into individual papers, it is possible that a page can get lost, which further increases the risks.
- As the only person in charge of on-spot data verification is the human operator, there is an increased risk of human mistake, e.g. writing down wrong vehicle number.
- Additionally, this increases the risk of intentional data altering - there exists no way to verify that the data has not been altered.

1.2 Research Questions

Regardless of whether the task is to create a digital solution that is based off an existing system or to create everything anew, it is crucial to conduct a research that will give insight about how the processes operate, or are meant to operate, at the moment. This will allow a faster and more informative way of laying down the foundations for the upcoming solutions.

Therefore, a set of goals was developed in order to gain as much information, which could be used in the practical solution writing, as possible:

1. Identify and describe business processes. This is an important step in understanding how the existing system works, providing information that could later be used for analysis, which gives understanding of which parts of the existing processes should be altered, in which way and why.
2. Analyze and construct the data model and data flow model. Data obtained from this research, provides crucial information about what data is transferred and allows building an electronic version of the same process. Furthermore, this particular goal serves as an additional tool to sort the business processes by their importance.
3. Design new data and application structures. This uses the information obtained from the previous two tasks as a means to lay a theoretical and abstract ground for the upcoming system. This task involves sketching the potential data types, database layouts and data relations as well as application design.

1.3 Preparation

The aim of the theoretical part, besides providing a foundation for the practical solution, is to get insight regarding the following subjects:

- Business process research. In order to develop an application for an existing business system, it is essential to understand what a business process is, how it operates and what its key elements, weak and strong parts, are.
- Technical task design. A challenge of forming a list of specifications and requirements is crucial for reshaping the client's view into technical terms.

- Study data models and data flows. This research field belongs to physical observations, as it requires understanding of how data is transferred within an existing business process. Information obtained from this research field allows a safer and more reliable vision of business process digitalization.
- Oracle database and APEX application server. The client insisted that the work should be done using Oracle tools. Therefore, it also becomes a research field of its own, as it is important to understand what an underlying system is, before using it as a tool.

2 Terms and Notations

Client (Estonian *klient*) - a representative word for the company Digiline OÜ that requested this solution. This term is present here for simplification and generalization reasons.

Object of improvement (Estonian *objekt*) - a representative word for the client'

Business process mapping (Estonian *äriprotsesside kaardistamine*) - a term which describes a set of tasks with the ultimate goal being understanding what a each user involved does, understanding their roles, responsibilities. The main goal of business process mapping is to provide organizations with information that could be used to increase its effectiveness.

Oracle APEX (Oracle Application EXpress) is a web-based software development environment that runs on an Oracle database.

BPMN (Business Process Model and Notation) is a graphical representation of a business processes in a business process model.

Bizagi BPMN Modeler is a freeware application to graphically diagram, document and simulate processes in a standard format known as Business Process Model and Notation. Additionally, it allows exporting its models as Images, Visio, XPDL or BPMN.

DDL (Data Definition Language) is a data structure specific syntax. Its purpose is similar to one of a programming language - to define the said structures.

PL/SQL (Procedural Language/Structured Query Language) is Oracle Corporation's procedural extension of Structured Query Language (SQL) and the Oracle relational database.

MVP (Minimum Viable Product) is a product that has just enough functionality to satisfy the initial customer requirements in order to provide feedback for future development.

3 Background

3.1 Research Methods

In order to develop a digitalization outline for business processes, it is essential to conduct research, in order to bring out the current state of the said processes. This is called business process mapping, which allows the outside companies to assess whether or not there exist any processes that can be improved and, ultimately, how it should be done.

This chapter is dedicated to describing what type of research was conducted in order to perform business process mapping, how it was done and describes the results. The final chapter 3.3 provides information about how those results can be interpreted and applied considering the limitations and requirements mentioned in the previous chapter, leading to the practical part chapter.

To assess and sketch the existing workflow of the client's business processes, the following methods were considered necessary and effective:

- Direct interaction with the client's representative. As there is a set of goals that should be achieved, the representative is the best option of obtaining high-level information about the infrastructure, such as the number of workers, their positions, the estimated number of workers.
- Interview with the actual workers on their workplaces. This particular method allows gaining an end-user perspective, which is vital to sketching a correct model. This is considered necessary to account the end users, as they are the people who will actually be using and assessing the developed solution. Furthermore, the end-users can provide an additional insight of what is considered practical and impractical in the current system, allowing even more options for changes that can improve the workflow.
- On-spot observation of the workflow process. It is considered a good practice to personally study the potential object of improvement, as it gives insight on the actual process details, rather than abstract or clouded answers from users. This particular method allows the observer to study not only the workflow itself, but also analyze its most vital and time-consuming points, as one of the goals of this project is to speed up the performance. Additionally, this gives the observer basic understanding of the user roles, their permissions and tasks, which later allows an easier and faster design of permission systems.
- On-spot observation of the document and data flow process. This specification is necessary, as currently certain operations require communications via cellular network. An example of this would be the exit procedure, when the operator verifies that the vehicle's stay is paid for. This is the second essential part to business process mapping, as it allows the observer to understand which data is transferred, when and between which participants, which participant roles have access to which documents and data and which participants can send which signals to others. This allows an easier data access restriction definition in during the development phase.

3.2 Research Results

For the ease of understanding, the results of the research topics will be split into four sections, each section corresponding to the topics from the previous section under the respective position.

- Direct interaction with the client’s representative. As this part is directly related to the restrictions and requirements of the project, it will be described in more detail in the next section.
- Interview with the actual workers on their workplaces. This interview provided information regarding the expected application and its functionality from the end-user perspective. This allowed altering some of the early design models to account several additional usability requirements. The interview was not structured, as it was deemed more efficient to ask additional questions when necessary, rather than repeating the same questions multiple times. The reason behind this is that the aim of the interview was not to get the average vision, but to bring out the most critical problems of the existing system.
- On-spot observation of the workflow, document and data flow process. The research on this particular topic has shown that such data transactions as entrance and departure forms, accounting documents, are stored using paper-based methods. Additionally, as already mentioned in the previous section, the observation and research has shown that, in addition to the initial assumption that paper-based system is used to transfer data. For that, cellular connection is used to verify the fact that the leaving vehicle and all trailers, if present, have been paid for by the respective registered company. As a result, the existing data flow is completed via 2 routes - paper and telephone.

The described observations provide enough information for creating process models (see Fig. 1, Fig. 2). These business process model representations, created in Bizagi Process Modeler, serve the purpose of being more understandable to both the developer and the client.

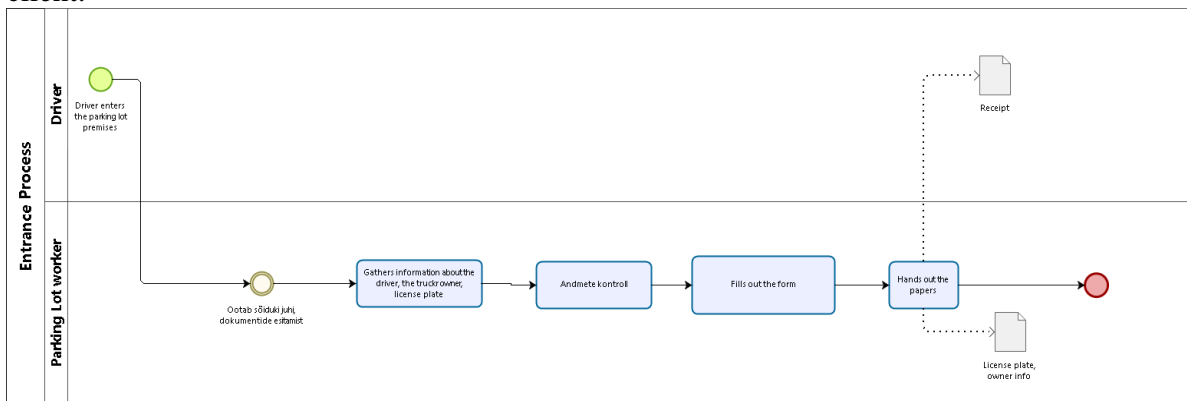
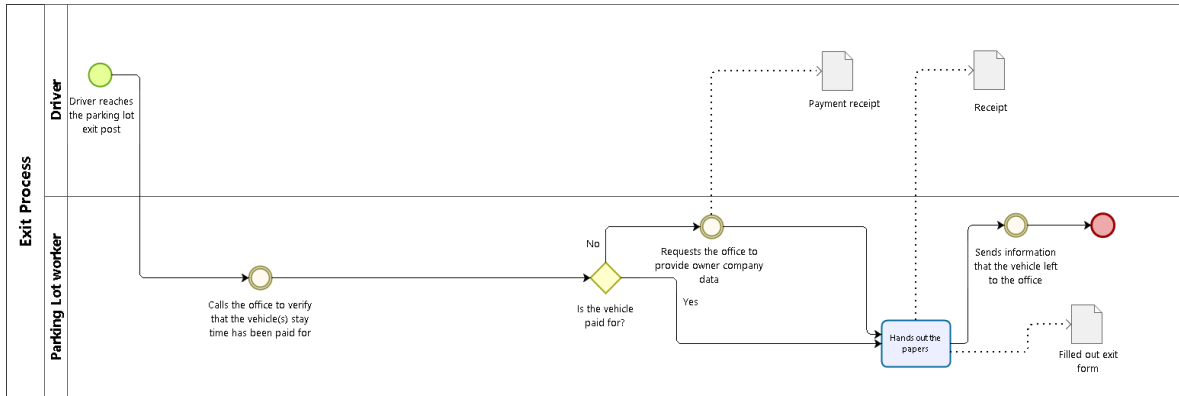


Figure 1. Process model showing the entrance event, designed based off the interview data.



Powered by
bizagi
Module

Figure 2. Process model showing the exit event, designed based off the interview data.

From those figures, it is clearly visible that most of the steps are performed by the operator, meaning that all these steps require a certain amount of time from the operator and, additionally, when performed several hundred times a day, increase the risk of mistake and slow down the overall process.

Restrictions and Requirements

During the interviews and the research, a list of restrictions was created. The major ones are:

1. The project is supposed to be capable of working on an existing software package - a virtual server.
2. Hardware limitations - 16GB RAM, 4 cores.
3. The changes to the existing workflow should be minimal.

Additionally, a set of requirements have been drawn out:

1. Because the solution is designed for people that do not possess good knowledge of PC and to avoid unnecessary time consumption, the client workplaces should not require any additional installations.
2. As the client has already dealt with virtual servers, they strongly insisted on using Oracle, as they have previous work experience with it.
3. The system is supposed to be used in an enclosed network, meaning it should not require any external connections. This requirement was drawn out for the purpose of decreasing the vulnerability of the system.
4. The general form of the solution should consist of an application-database tandem. Such solution is preferred as it is both a standard and allows a safe, reliable and understandable environment.

3.3 Architecture and Structure

Based on the restrictions and requirements stated in the previous part and taking into consideration the research results, the following solution was designed.

The system should consist of 2 parts - a web application, that the end users are supposed to be using; and the database, where all the data is stored. The development environment for it was chosen to be Oracle XE, as it has several advantages over its alternatives:

- Oracle XE provides a free version, which has a storage limit of 11 GB, which is sufficient for storing all the data for this application for at least several years [3]. Such time frame is sufficient to assess the quality of the application and decide whether or not a paid version is required. Lastly, such time span is sufficient for logging information, as it should not affect short-term operations.
- Oracle XE provides an already existing system of a database and an application development system (APEX), which simplifies both the connection establishment and security verification processes [4].
- Oracle XE provides an easy access to both file system as well as most of currently popular databases, which provides opportunity for further integration.
- Oracle XE offers both visual and raw code editor, effectively allowing a combination of visual programming and raw programming. This aspect was considered, as it is obvious that designing and ordering elements of the page is easier done using the visual editor, as it is possible to know in advance, what the visual output will be. On the other hand, writing database queries and behaviour patterns is easier done using the code editor, as it allows to avoid limitations that the visual environment has.
- Oracle APEX is a high-grade application server, which also serves as a development environment, which provides both a sufficient level of security, which includes authentication and user authorization protocols, and ease of access [5].

As the primary system that was used in the practical part of this paper is Oracle APEX, a more detailed review of it is provided below.

APEX

APEX can be used to build complex web applications that can be used in most modern web browsers. The APEX development environment is also browser-based.

APEX is used internally by Oracle to develop some of its support sites. The AskTom knowledgebase and online store both run on APEX. Oracle Corporation provides a support system that is constantly being improved - My Oracle Support [7].

APEX advantages are:

- DBAs familiar with PL/SQL can use their skill set to develop web applications
- Easy to create mock-ups using pre-built themes
- Easy to deploy (end user opens a URL to access an APEX application)
- Scalable (can be deployed to laptops, stand-alone servers, or Oracle RAC installations)
- Server-side processing and validations
- Basic support for group development
- Free hosting of demo applications provided by Oracle
- Apex applications can run on the free Oracle Express Edition (XE) database
- Individual components of an application can be retrieved or identified using SQL, facilitating customized reports
- Easily adheres to the SQA development/test/production model (while not exposing DB passwords)
- Helps put the focus on the DB model supporting a solution (versus coding in Java, .NET or PHP - you only need JavaScript)

- Easily supports a standardized theme across application sets (and the changing of that theme)
- Included changes tracking and code documentation
- Included group working tools

4 Implementation

4.1 Solution Description

This thesis chapter describes the practical implementation of the solution described in part 3, giving insight on the general progress workflow, the issues encountered and their solutions. The practical implementation of the development of MVP is divided into 3 chapters. First one is database development, which covers the information about Oracle database and how it was utilized. Second stage is about web application development, as it describes the development process of graphical user interface. Finally, the last chapter is about business logic development, it covers the layer between the previous two elements - Stored Procedures.

The implementation is based off the information obtained from the system analysis, as well as additional information, obtained from sprints.

4.2 Specification of Business Processes

After analyzing the business processes and their flow at the present time, another set of process models was developed in order to formulate and visualize the work that should be done in order to develop an application.

As seen on Figures 3 and 4, the new models, which are designed for the application, move most of the job that the parking lot worker currently does to the system. The job of the worker is simplified as much as possible, while leaving also room for action. For example, while the system will automatically scan the license plates, the worker will still be able to verify it physically and alter the information, when needed. However, if the vehicle has been registered in the database before, the data regarding that vehicle will be offered to the worker. This data can include the vehicle's owner, data regarding the vehicle itself, such as the registered country or classification.

Finally, as seen on Figure 5, the new system will allow the accounting to cease manual requests and searches for information regarding each contractual client. Instead, all information about the amount of time those clients' vehicles have spent on the parking lot premises within the month is sent to the accounting automatically each month.

It is also worth noting that it will still be possible to request these reports at any other time, however the purpose of a timed event is to automatize the action and relieve the accounting from additional unnecessary actions.

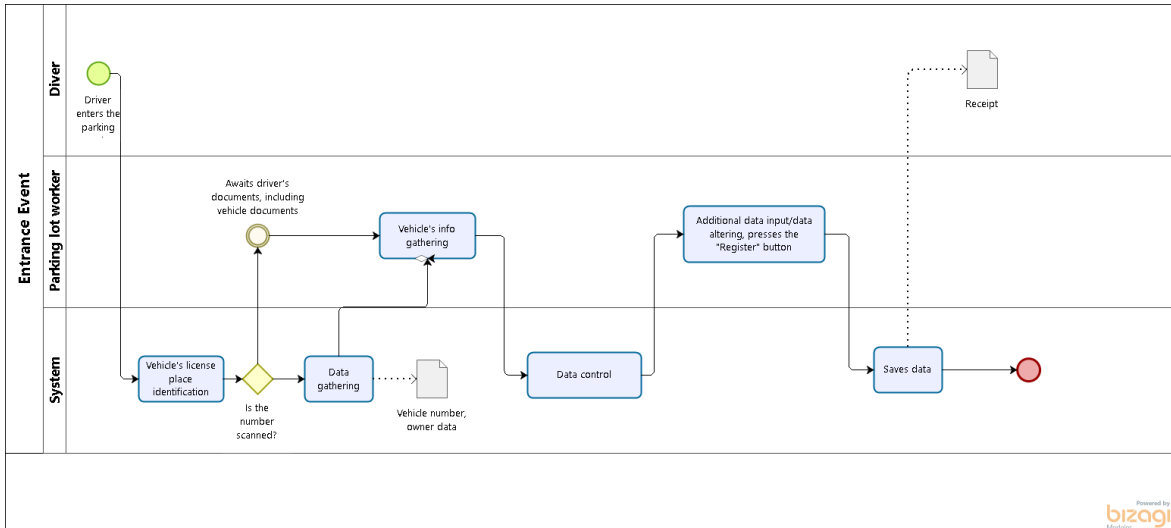


Figure 3. Process model showing an altered version of the entrance process, involving the system.

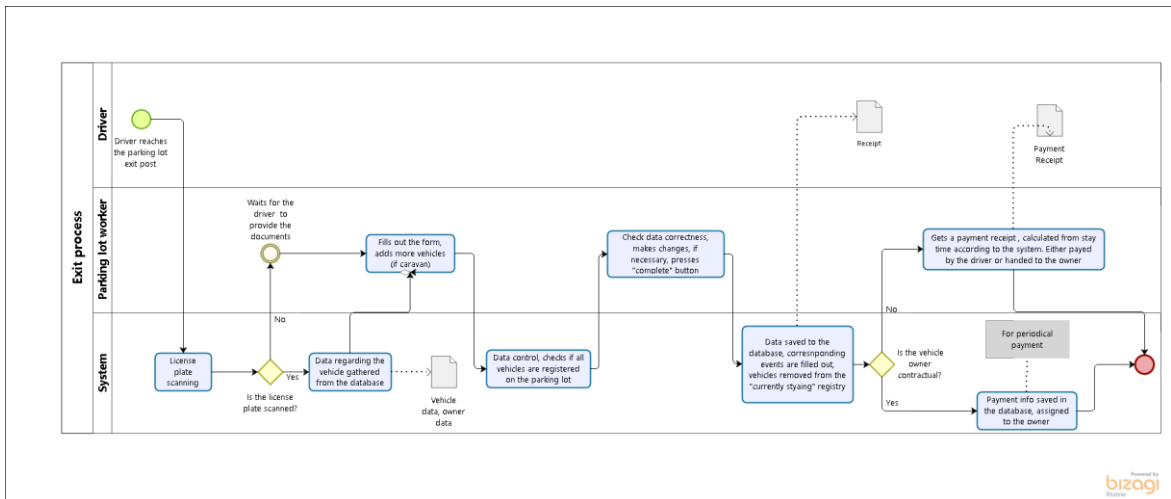


Figure 4. Process model showing an altered version of the exit process, involving the system.

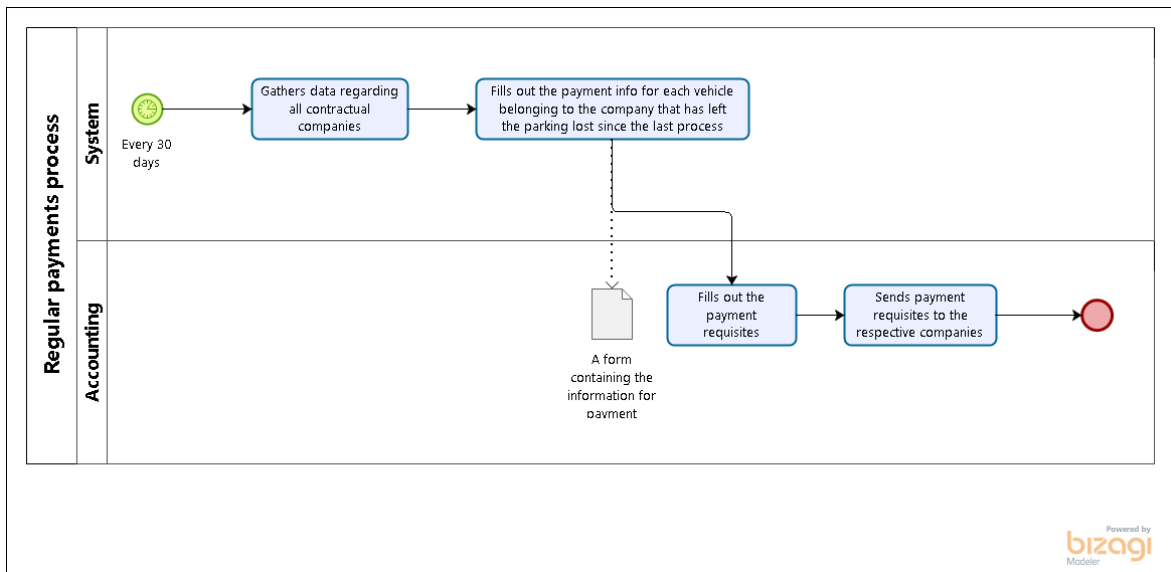


Figure 5. Process model showing a timed event where system provides the accounting with payment information to be sent to the contractual vehicle owners

4.3 Database Development

As the development process requires adaptation to the situation, during it the database shape has changed several times.

After the research was completed, the database design was made to help visualize the task at hand and to solidify the key elements of the database. Figure 6 shows that initially the database was supposed to consist of 8 core tables, which are:

- Users. This table contains the information about the registered users and their permission level. Users with the role “Superuser” can add new users and assign roles to them, while unauthenticated users cannot access the application. Additionally, this table stores information about User’s last login time, allowing an easier track of potential break-ins. In the future version, it is planned to add an additional table that will store all user login dates. User can have only one User Role to avoid inconsistencies. Finally, when registering a user, one can provide a contact email. Currently, this serves no purpose besides the contact link with the physical user. In the future, however, this function can be used to either notify the user that a login was made.
- User_Roles. Describes the roles of the user. Depending on which role User has, their permissions and access to different functions of the application are limited. For example, User with user role “Operator” has no rights to access application settings, while a User with “Superuser” role can add, alter or remove user roles.
- Event_Type. This table contains entries that describe application event types. For the thesis, only “Entrance” and “Exit” events are accessible and necessary.
- Event. This table holds information about an event, such as what type of event happened, at what time.
- Event_Transport. This table provides additional information to Event table, as it shows vehicles that correspond to each event. Additionally, Number_in_column column provides understanding in which order the vehicles entered or left the parking lot premises. Note that this field can be filled out only if there is a caravan of vehicles, consisting of a tractor and up to several trailers.

- Transport. This table serves as a reference point for Event_Transport. It contains a detailed information about all vehicles that enter the parking lot area. As a result of the interviews and research, only the information regarding the vehicle number, owner company and transport type was deemed necessary.
- Transport_Owner. A table that contains the names of all known vehicle Owners. As the object is meant for business vehicles, it is expected that most of the vehicles will belong to companies, rather than private owners.
- Transport_Type. This table only contains information regarding the types of vehicles. For the application development process, 3 values were added - Tractor, Trailer and Van. Tractor type corresponds to the head of a vehicle column, to which trailers are attached. For the purposes of testing, van type was introduced as a means to differentiate all the remaining vehicle types.

As it is important for both development and exploitation phases to log all events, it was considered necessary to include 4 additional columns to each table:

- Created_At, Created_by. Values to these columns are assigned using triggers and are immutable at after the creation of a new database row. They provide information about when each database row was added and which user made the entry. Aside from basic logging, this also provides some basic security.
- Updated_At, Updated_by. Values of these columns are updated using triggers, when the corresponding database rows are changed. As the user cannot directly modify these columns, they serve as additional security measurement, reducing the chances of unauthorized or anonymous data modification.

It is worth noting that the user that creates or modifies the entry can be ether application or database user, meaning that database row modification is tracked and recorded for both user types. Figure 7 shows database rows, where there are both freshly created rows and ones that were modified by various users, where ADMIN is the user of the application and TEST is the database user.

In the process of agile sprints/meetings with the client, a certain amount of additional requisites was drawn out, such as:

- For tables USERS, TABLE_VEHICLES, formerly Transport, and TABLE_COMPANY, called Transport_Owner in the theoretical model, columns VALID_FROM and VALID_TILL were added. These columns act as restrictions to limit access to the corresponding data. For example, if a certain user's VALID_FROM date is 13.06.2020, then that user is inaccessible until that date.
- COUNTRIES table was added, to assist the TABLE_VEHICLES table. It establishes connection between country names and their 3-letter codes (ISO_ALPHA3), according to ISO-3166 international standard [8, 9]. This serves as an additional information about a vehicle origins, additional to the license plate number.
- For TABLE_EVENTS table, formerly EVENTS, EVENT_STATUS column was added. This column is designed to hold the state of an event. For example, 'D' status indicates that the event has been deleted. Because of that, there is no necessity to really delete database rows, which both improves security and provides additional information for logs. Possible values are currently restricted using constraints, rather than adding a new reference table as there are only 2 acceptable values besides 'D' - 'R', which stands for "ready", or 'P', which stands for "processing".
- For table USER_ROLES, column SHORTNAME was added. It contains a formalized version of the role name. In difference from standard role name, which is visible

to all application users, SHORTNAME is a name used by application code, as it is more understandable to write a role shortname, rather than role ID.

Additionally, it is worth noting that the current database state contains only the minimum amount of requisites required for the initial exploitation launch. In other words, the minimum required for accounting. The database structure is supposed to be developed during the upcoming agile sprints.

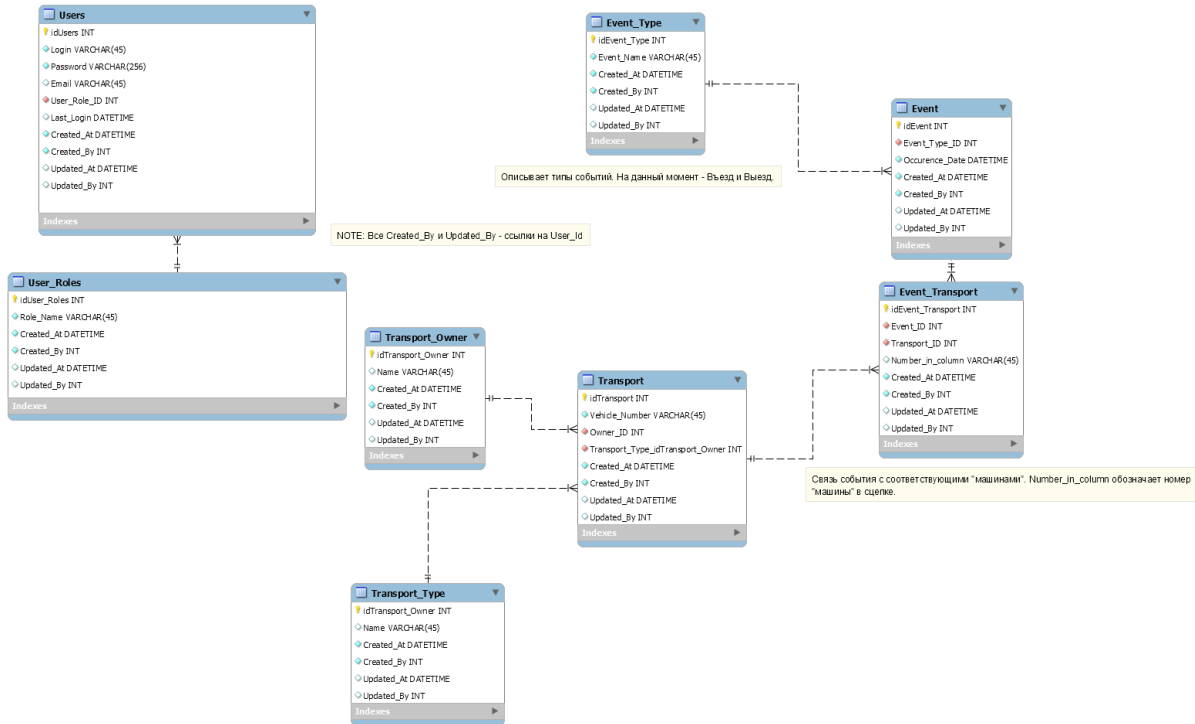


Figure 6. An initial model of the database, which was sent to the client. Some notes are in Russian.

TABLE EVENTS										
Table	Data	Indexes	Model	Constraints	Grants	Statistics	UI Defaults	Triggers	Dependencies	SQL
Query	Count Rows	Insert Row								
EDIT	ID	EVENT_DATE	EVENT_TYPE	CREATED_AT	CREATED_BY	CHANGED_AT	CHANGED_BY	EVENT_STATUS		
	101	05/11/2018	3	11-MAY-18 01:54:20.511066 PM	ADMIN	-	-	P		
	26	05/04/2018	2	04-MAY-18 11:44:26.081528 PM	ADMIN	06-MAY-18 12:54:20.170920 AM	TEST	P		
	27	05/14/2018	2	04-MAY-18 11:48:15.516825 PM	ADMIN	06-MAY-18 12:54:22.587715 AM	TEST	P		
	28	05/14/2018	2	04-MAY-18 11:52:24.826182 PM	ADMIN	06-MAY-18 12:54:23.743706 AM	TEST	P		
	29	05/04/2018	2	04-MAY-18 11:54:01.394221 PM	ADMIN	06-MAY-18 12:54:25.542633 AM	TEST	P		
	30	05/04/2018	2	04-MAY-18 11:55:21.272425 PM	ADMIN	06-MAY-18 12:54:26.731702 AM	TEST	P		
	31	05/05/2018	2	05-MAY-18 12:34:17.433027 AM	ADMIN	06-MAY-18 12:54:27.496445 AM	TEST	P		
	41	05/06/2018	2	06-MAY-18 12:01:12.450215 AM	ADMIN	06-MAY-18 12:54:28.456483 AM	TEST	P		
	42	05/06/2018	2	06-MAY-18 12:10:32.171491 AM	ADMIN	06-MAY-18 12:54:29.267440 AM	TEST	P		
	44	05/06/2018	2	06-MAY-18 12:55:00.828994 AM	ADMIN	06-MAY-18 12:59:15.586567 AM	ADMIN	R		
	61	05/06/2018	2	06-MAY-18 04:03:12.350417 PM	ADMIN	-	-	P		
	81	05/09/2018	2	09-MAY-18 01:14:09.936735 AM	ADMIN	-	-	P		
	25	05/04/2018	2	04-MAY-18 11:23:17.656631 PM	TEST	06-MAY-18 12:54:30.091370 AM	TEST	P		
	32	05/05/2018	2	05-MAY-18 12:50:18.629425 AM	ADMIN	06-MAY-18 12:54:37.451233 AM	TEST	P		
	33	05/05/2018	2	05-MAY-18 01:33:42.909962 AM	ADMIN	06-MAY-18 12:54:41.531146 AM	TEST	R		

Figure 7. Data rows in the EVENTS table. It is visible that certain rows do not contain any values in “CHANGED_AT” and “CHANGED_BY” columns

4.4 Web Application Development

Another part of the development process involved creation of a graphical interface for the end user. For that, App Builder, which is a built-in tool in APEX, was used. APEX provides 2 ways of application editing - a visual programming interface and a template editor.

Visual programming interface allows the user to quickly and efficiently assemble pages from existing parts. All pages are divided into several parts:

- Pre-rendering. A part where computations and processes are stored. This part of the page is executed by APEX before the page is rendered to the end user. All of the preparation work, such as filling out the page fields, making queries and such, are executed here.
- Regions. This part corresponds to what the page actually consists of. It describes the page’s header, breadcrumbs, body etc. In this section, all the template elements are added.
- Post-rendering. This part is executed only by APEX. Usually, processes and computations that describe page transactions are stored here.

Template editor is the secondary tool of the APEX engine. It gives the user means to customize the parts that are sent to the page. While it was mentioned that Regions section consists of a set of templates, it is also worth noting that templates describe all APEX-made pages. For example, there are specific templates that define page header or body. A set of templates can be joined into a Theme, which can be applied to a page without the necessity to manually write page code every time.

While this means that it is impossible to directly alter the page source code, the combination of visual programming interface and template editor completely replace this necessity, providing a simpler editing method without restricting the functionality.

4.5 Business Logic Development

The native programming language in Oracle environment is PL/SQL. This language is simple in syntax, yet it offers full functionality, including object-oriented programming and other modern tendencies in programming. The best feature of PL/SQL programming language is very tight integration with database-access constructions (SQL phrases).

Part of application logic is stored as Stored Procedures in the database, while others are stored as PL/SQL code blocks, embedded in the logic of web application’s pages. Because Stored Procedures are contained within the database, they provide a more secure approach than standard PL/SQL code blocks. Furthermore, Stored Procedures allow business logic encapsulation, as they allow embedding business logic in the database. Furthermore, Stored Procedures, unlike web application-embedded code blocks, act as existing functions, programs or procedures. This implies that they can be accessed and used multiple times from various parts of the application without the necessity to rewrite the logic.

Furthermore, Stored Procedures are more efficient when considering their execution time. Because they are contained within the database itself, they do not require neither time nor resources for the connection to the database.

Finally, because Stored Procedures can be written as complex sets of commands, they allow verification of data integrity, while also performing data modifications. It differs from standard PL/SQL code blocks, as Stored Procedures can take way less input parameters to initiate a sequence of commands, lowering the chance of attacks via SQL queries.

Based off these points, it was decided that most of PL/SQL logic should be moved to the Stored Procedures. PL/SQL code blocks are still used within the application itself, however, their usage and invocation has been reduced to a minimum. Figure 8 shows the an example of how Stored Procedures are used within an APEX code block. It is visible that no actual function definition is made and only the results are handled. In this particular case, Stored Procedure *check_vehicle_resistance* is used to check the presence of a vehicle on the object.

▼ Identification	
Name	Entrance presence check
▼ Execution Options	
Sequence	10
▼ Validation	
Editable Region	- Select -
Type	PL/SQL Function
PL/SQL Function Body Returning	
Error Text	
<pre>BEGIN IF check_vehicle_existance(:P3_EVENT_DATE, :P4_VEHICLE_NUMBER) THEN RETURN NULL; ELSE RETURN 'The vehicle is already present on the parking lot!'; END IF; END;</pre>	
Always Execute	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No

Figure 8. Stored procedure *check_vehicle_resistance* implementation.

4.6 Version Tracking

APEX provides built-in features for tracking changes made in the code. These feature include, but are not limited to, tracking the time the changes were made, by whom, what type of change was made (e.g. a page was created, changed or deleted) or which component was changed.

This feature is important when a cooperative work of multiple developers or developer teams is considered, as it not only provides log information, but also decreases the risk of intentional sabotage of the application.

Additionally, APEX provides another feature, which is important when working on complex projects, which is the “Flashback export” option, which gives an option to restore the application, its page or components to one of the previous versions. This function relies on Flashback Query technology and allows you to restore the selected page to its state up to several hours ago [6]. The time limit is defined by the size of the tablespace that holds the information about the page states.

The limitation of the Flashback Export option is, however, its time limit, as it does not allow recovering the state of the page that is older than a certain, often unknown, time limit. While APEX by default does not offer this functionality directly, it provides a large set of so-called “Packaged Apps”, which is a storage of different apps that APEX supports. One of them, APEX Application Archive, allows exactly this.

4.7 Reports

As a means to provide feedback for analysis and, additionally, deliver more understandable information to the accounting segment of the client, a set of reports was introduced.

A report in APEX is a specific region of a web application page, which receives information from the database with the help of SQL queries. When invoked, be that upon the page loading or a specific in-page trigger, the report makes an SQL request to the database, fetching its data.

As an example, vehicle transfer report page will be described. As seen on Figure 9, the report consists of a table, which takes 2 dates as its time limits. After that, an SQL query is processed, which returns a list of all vehicles that have left the parking lot premises within the specified period of time. Additionally, it calculates the amount of days the vehicle stayed on the parking lot premises, which is always rounded up.

Owner Ts	Vehicle number	Entrance	Departure	Time of stay (in days)
ООО Фирма 1	111ABF	05.05.2018 00:03	14.05.2018 16:03	10
ООО Фирма 1	999BCD	06.05.2018 00:48	13.05.2018 18:54	8

Figure 9. User perspective of the report, the time frame is set to May 2018.

However, the reports also allow data manipulation. As seen on Figure 10, it is possible to alter the data of a database row by clicking on the edit button. This implies that reports can be used to not only deliver the data stored to the end user in an understandable format, but also provide an intuitive way of editing it as well, without giving too much freedom to one.

Shortname	Role name ru	Role name ee	Role name en	Comments	Active users	Changes info
DATA_ENTRY	Оператор	Operator	Data Entry	-	0	Данные впервые введены: SYV @ 30.04.2018 14:57. Последнее изменение: TEST @ 01.05.2018 16:55
USERS_ADMINISTRATOR	Администратор прав доступа	Kasutajate haldaja	User Administrator	-	0	Данные впервые введены: SYV @ 30.04.2018 14:58. Последнее изменение: ALEX @ 02.05.2018 11:49
SUPERVISOR	Начальник	Juhendaja	Supervisor	-	0	Данные впервые введены: SYV @ 30.04.2018 14:59. Последнее изменение: TEST @ 01.05.2018 16:55
SUPERUSER	Администратор приложения	Rakendusadministraator	Superuser	-	0	Данные впервые введены: SYV @ 30.04.2018 14:59. Последнее изменение: TEST @ 01.05.2018 16:57

Figure 10. An editable report showing a list of user roles and their information.

4.8 Testing

The aim of this thesis is to develop a minimum viable product, which will then be continuously and iteratively improved and developed further. However, no testing results are accessible by the moment of this paper's submission, as the project will be handled over to the client for testing by the end of May. The results of the testing will be used for analysis and future references. The main goal of testing will be to update the database with actual data, such as more diverse event, company and vehicle data. Such data is essential for understanding average occurrences of vehicles, their types, etc. Finally, such testing will provide feedback on the existing graphical user interface, as it must be both easy to understand and easy to use for the end user. All these actions will be performed as the next sprint of the agile development process.

5 Conclusion

The purpose of this thesis was to develop an enclosed-network web application for a business parking lot or, to be more precise, an MVP of it. The aim of the application was to digitalize the existing business processes for the client, as well as simplify the work of the employees, as they had to make many unnecessary and time-consuming steps in order to complete tasks that would take seconds for a machine to compute. This application provides a minimum required functionality to cover the basic operations - the registered users can only access the system, registered users can register entrance and departure events for vehicles, users can get reports on the events within specific date or a date frame. There exist restrictions that prevent users from intentionally tamper with stored data in form of Stored Procedures and verifications.

The author also conducted research in the requirements analysis field. He held personal interviews and observations of the workflow and business process, gaining insight on the process from both the employer and employee perspective. This allowed him to develop a set of guidelines that would satisfy both sides. For example, the event registration can be accessed in two ways, which does not affect the employer in any way, however speeds up the employee's work speed, as both ways fill out certain fields for them.

At the end of this thesis, author describes the project's potential and lists steps that need to be done in order to turn the application into a fully functional product.

By the of this paper's submission, client's feedback cannot be provided as the product will be handed over for initial testing only by the end of May or in June.

Because the aim of this project was to provide a MVP by the deadline of the thesis, there exist a vast field for improvements, which will be described in more detail in the next section. However, it is worth noting that the author cannot provide a full list of improvements that can or will be done, as new goals may arise as a result of agile sprints to come.

6 Project Potential

6.1 Transfer all Queries into the Database

For both speed and security reasons, all PL/SQL queries that currently remain in the application code block converted into Stored Procedures in the database. This means that the end user will have no direct interaction with the database, but will interact with the intermediate API layer instead.

6.2 Automatic Reports

A set of reports will be developed that will provide daily/weekly/monthly information about the application's performance to, for example, accounting. This implies that accountants will simply have to send the already prepared forms to the corresponding people.

6.3 Additional System Integration

It is essential to make the application capable of processing data from internal systems that are already present on client's side. The most essential is the license plate scanning system. When the application can read that data and fill in the forms itself, it removes some of the workload off the operator, leaving him only with the task of verifying the data and adding trailer data, should it not be present.

6.4 User Role Differentiation

Users will have separate roles that will restrict their access to certain data or sections of the application. For example, a user with "Operator" role will be unable to view the application settings, while one with "Superuser" role can.

6.5 Obtain Extensive Data for Analysis

As the initial version of the application will be deployed into the testing, the database will be filled with a sufficient amount of data, which by would far exceed what can be added for testing purposes. This allows an extensive analysis of the tendencies within the transport and additionally provides the feedback regarding the application state from the end user perspective.

7 References

- [1] Haarmann, Harald (2002). “Geschichte der Schrift”, DE: C.H. Beck.
- [2] Steinberg, S. and Warde, B. (1974). Five hundred years of printing. 3rd ed. Harmondsworth, Eng.: Penguin Books, pp.19-28.
- [3] Piotrowski, P. (2018). Oracle Database 11g Express Edition Quick Tour. [Online] Oracle.com. Available: <http://www.oracle.com/technetwork/articles/sql/11g-xe-quicktour-498681.html> [Accessed 12 May 2018].
- [4] Docs.oracle.com. (2018). Oracle Application Express Release Notes, Release 5.1. [online] Available at: <https://docs.oracle.com/database/apex-5.1/HTMRN/toc.htm#HTMRN-GUID-E4353E38-3250-4304-8982-F6D4A7887227> [Accessed 12 May 2018].
- [5] Knox, D., Gaetjen, S., Jahangir, H., Muth, T., Sack, P., Wark, R. and Wise, B. (2009). Applied Oracle Security. New York, USA: McGraw-Hill Professional Publishing.
- [6] Czarski, C. (2018). Recover accidentally deleted APEX components: Flashback Export. [Online] Apex.oracle.com. Available: <https://apex.oracle.com/pls/apex/german-communities/apexcommunity/tipp/5921/index-en.html> [Accessed 12 May 2018].
- [7] Author, G. (2018). Introducing... My Oracle Support!. [Online] Blogs.oracle.com. Available: <https://blogs.oracle.com/supportportal/introducing-my-oracle-support> [Accessed 12 May 2018].
- [8] United Division (2018). UNSD — Methodology. [Online] Unstats.un.org. Available: <https://unstats.un.org/unsd/methodology/m49/> [Accessed 13 May 2018].
- [9] Iso.org. (2018). ISO 3166 Country Codes. [Online] Available: <https://www.iso.org/iso-3166-country-codes.html> [Accessed 13 May 2018].

8 Appendix

I. License

Non-exclusive licence to reproduce thesis and make thesis public

I, **Stanislav Belogrivov**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

of my thesis

Development of Web Application for Business Parking Lot,

supervised by Siim Karus

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, **14.05.2018**