

OutSystems as a Rapid Application Development Platform for Mobile and Web Applications

LAHTI UNIVERSITY OF APPLIED
SCIENCES
Degree programme in Business
Information Technology
Bachelor's Thesis
Spring 2017
Dmitry Golovin

Lahti University of Applied Sciences
Degree Programme in Business Information Technology

GOLOVIN, DMITRY:

OutSystems as a Rapid Application
Development Platform for Mobile and
Web Applications

Bachelor's Thesis

40 pages

Spring 2017

ABSTRACT

OutSystems is a visual low-code development environment targeted at citizen developers. It is relatively easy to create and deploy fully featured applications with OutSystems even for a novice developer.

The purpose of this study is to determine the main problems that the students of Lahti University of Applied Sciences experience and find out how these problems can be solved with OutSystems.

The theoretical part of this study explains the choice of OutSystems as a Rapid Application Development Platform for creation and deployment of Mobile and Web applications and explains the basic concepts behind this technology.

A survey and several interviews were conducted for data collection for this study. A mobile application and a web application implementing popular student services were developed as a part of the case study.

Based on the conducted research, the conclusions about the preferred order and the optimal methods of development of the mobile applications for the student services were made. The suggestions about the design from the case study should also be taken into consideration in order to build an application to be accepted by the majority of the users.

Keywords:

OutSystems, Rapid Application Development, student services, Mobile Applications, Web Applications, Platform as a Service

TABLE OF CONTENTS

1	INTRODUCTION	1
2	RESEARCH QUESTION	2
2.1	Reasons for this study and research questions	2
2.2	Research methods	3
2.2.1	Surveys – the source of quantitative data	3
2.2.2	Predictive extrapolation	4
2.2.3	Case study – development of two applications	4
2.3	The study in short	5
3	LITERATURE REVIEW	6
3.1	Mobile applications vs. Web applications	6
3.2	Student services of Lahti University of Applied Sciences	7
3.3	Previous survey	7
3.4	Trends in mobile device usage	8
3.5	Popularity of mobile devices in Finland	8
3.6	OutSystems development platform	10
3.7	Rapid Application Development	11
3.8	Internal structure of OutSystems	11
3.9	Supremacy over competitors	12
3.10	Accessibility guidelines compliance	13
3.11	OutSystems used by higher education institutions	14
4	DATA ANALYSIS	15
4.1	The survey	15
4.2	Comparison with the previous survey	16
4.3	The interviews	18
5	THE CASE STUDY	20
5.1	OutSystems as a Development Platform overview	20
5.1.1	Steps to create an application with OutSystems	20
5.1.2	Development environment	23
5.1.3	Customizations and handling crashes	28
5.2	Alternative web application for Menus & Lunchtimes	28
5.3	Android and iOS application design for Lukkarikone	31
6	CONCLUSIONS	34
7	SUMMARY	36
	LIST OF REFERENCES	37

LIST OF KEY TERMS AND ABBREVIATIONS

.NET – a framework developed by Microsoft as an abstraction layer above the operating system to bring software on a higher level and make it more portable

5G – a set of future standards describing mobile networks and wireless communications, focused on increasing speed and reducing latency of current standards

Agile software development – a software development methodology with flexible requirements targeted at rapid application delivery

Android – a mobile operating system developed by Google

Application programming interface (API) – a description of methods that an application expose to be used by third-party developers

Blackberry – a retired mobile operating system developed by BlackBerry Limited

Bring your own device (BYOD) – an initiative encouraging the students and staff to use their own personal computers and mobile devices promoted at Lahti University of Applied Sciences

Business Application Programming Interface (BAPI) – an API provided by SAP enterprise resource planning software

C# – a high-level programming language designed to run in a virtual machine developed by Microsoft for the .NET platform

Cascading Style Sheets (CSS) – a declarative language for describing the visual style of a graphical user interface, one of the three core technologies of web applications

Citizen developer – a regular user without any programming knowledge or understanding of computing principles who is able to create business applications to be used by the others

Client-server model – a typical communication pattern for the Internet where one of the sides is a master and the other is a slave

Clipboard – a buffer for data to be stored and transferred across different applications within a single computer

Cloud computing – a concept from the 70s describing a matrix of interconnected computers providing access to data and calculations of the whole system as a single unit, now becoming a reality

Common Language Runtime (CLR) – the virtual machine used by the .NET platform

Debugging tool (debugger) – a computer program allowing to find bugs in software by executing it step-by-step and analysing the memory and the stack

Eclipse – an integrated development environment developed by the Eclipse Foundation

Extensible Markup Language (XML) – a markup language to form data in a strict text form that is readable by human and parsable by computer

Facebook – a social networking website that has official and unofficial pages of Lahti University of Applied Sciences

GNU – an operating system developed by the GNU Project used on servers, desktops and mobile devices

HyperText Markup Language (HTML) – a markup language used for creating web pages, one of the three core technologies of web applications, the latest version is HTML5

Integrated development environment (IDE) – a computer program for software development providing an easy to understand interface and allowing to start development without understanding of underlying concepts

IntelliJ IDEA – an integrated development environment developed by JetBrains

Internet of Things (IoT) – a concept from the 80s describing microcomputers embedded into various appliances and interconnected with each other in order to exchange information, now becoming a reality

iOS – a mobile operating system developed by Apple

JAR – a file format for archived Java packages used to deploy Java applications

Java – a high-level programming language designed to run in a virtual machine developed by Sun Microsystems now acquired and maintained by Oracle

Java Platform, Enterprise Edition (J2EE) – a variation of the Java Runtime Environment targeted at enterprise level businesses

Java virtual machine (JVM) – the virtual machine of the Java Runtime Environment

JavaScript – a high-level programming language, one of the three core technologies of web applications, the latest standardized version is ECMA-262 (ECMAScript 2016)

JavaScript Object Notation (JSON) – a data format used to transmit data between different applications in a form of text

JBOSS (now WildFly) – a J2EE application server developed by Red Hat

Keyboard focus – an indication of an active area on the screen where the text inputted from the keyboard appears, usually highlighted graphically and often has a cursor

Linux – a kernel of various operating systems, used by Android and often GNU, however not an official kernel of GNU

macOS (formerly OS X) – a desktop operating system developed by Apple

Operating system (OS) – a set of computer programs providing an API for developers as well as some sort of user interface for the users

Platform as a Service (PaaS) – a platform hosted in a cloud with provided software and hardware with little to no flexibility

Rapid Application Development (RAD) – a software development methodology suiting UI driven software development process targeted at fastest possible deployment of the applications

Representational state transfer (REST) – a method of intercomputer communication usually utilizing JSON to format the data

SCRUM – a sprint-based project management technique most suitable for small to medium software development team

Simple Object Access Protocol (SOAP) – a method of intercomputer communication utilizing XML to format the data

Symbian – a retired mobile operating system developed by Nokia

ThingWorx – a Rapid Application Development Platform developed by PTC

Uniform Resource Locator (URL) – a unique string identifier of any resource on the Internet

Visual Studio – an integrated development environment developed by Microsoft

VK – a social networking website that has an unofficial, yet popular page of Lahti University of Applied Sciences

Web browser – a computer program for displaying web pages from the Internet

Web Content Accessibility Guidelines (WCAG) – a set of guidelines on creating web content acceptable for people with disabilities and/or highly limited devices, such as mobile phones

WebLogic – a J2EE application server originally developed by BEA Systems now acquired and maintained by Oracle

Windows – a desktop operating system developed by Microsoft

Windows Phone – a mobile operating system developed by Microsoft

Yammer – a social networking service officially used and promoted by Lahti University of Applied Sciences

1 INTRODUCTION

Development of mobile and web applications is becoming more and more popular because of two factors: the ongoing shift from personal computers to mobile devices and switch from native applications to web browser based alternatives. Moreover, many organizations around the world are now adopting cloud computing solutions, and massive calculations are done on servers rather than on personal devices of the users. This study focuses on the development of mobile and web applications using cloud-based platforms, in particular OutSystems.

OutSystems is a visual low-code development environment that comes in two parts: a server and a desktop application for developers. The desktop application is used to design mobile and web applications with simply dragging and dropping. The platform allows to easily model databases, workflow and rules and then deploy to production server or multiple digital distribution platforms with just one click. (OutSystems 2017e.)

Lahti University of Applied Sciences was established over 25 years ago and became a limited company in 2015 (LAMK 2017a; 2017b). According to a comparison made by Talouselämä, it is in the list of top 10 Finnish universities of applied sciences (Mikkonen 2013). There are about 5,000 students currently studying in the university (LAMK 2017b).

The students of Lahti University of Applied Sciences currently use several applications (often referred to as student services) needed for their studies, but student services turn out to be not optimized for use on mobile devices. This study suggests several general improvements that can make student services more usable.

Lahti University of Applied Sciences has recently licensed OutSystems. OutSystems is already used by many companies including other educational institutions, and the servers running it are located all around the world. Lahti University of Applied Sciences can benefit from the use of OutSystems by developing applications for its students and the teachers.

2 RESEARCH QUESTION

This chapter defines the reasons why this study is important, formulates the research questions, describes the research methods and the reasons for choosing them, the limitations and the structure of this paper.

2.1 Reasons for this study and research questions

This study is important because of a variety of reasons. Mobile devices have become increasingly popular, and students of Lahti University of Applied Sciences have been dissatisfied with how the student services work on mobile devices. OutSystems, of which the university recently purchased a license, can be used to create fully-featured and mobile-optimized web and native mobile applications with minimal effort. This study shows how the university can possibly benefit from OutSystems by developing applications that do not have the problems that the students have complained about.

One of the most important ideas behind this paper is the concept of the citizen developer. A citizen developer is “a user who creates new business applications for consumption by others using development and runtime environments sanctioned by corporate IT” as defined by Gartner IT Glossary (2017). With cloud-based platforms like OutSystems it is possible for virtually anyone to create fully-featured business applications with little to no programming.

Finally, this study may be useful for software development teams at Lahti University of Applied Sciences as well as independent developers. The main research questions of this study are:

- What problems do the students of Lahti University of Applied Sciences experience with existing student services on mobile devices?
- How can OutSystems answer to these problems?

2.2 Research methods

Multiple research methods are used in this study, because multimethodology has been proven to be more efficient than the monomethod approach (Davis, Golicic & Boerstler 2011, 473). Three main methods of research were chosen: a survey, statistical extrapolation and a case study.

2.2.1 Surveys – the source of quantitative data

A survey of 14 questions was conducted among the students of Lahti University of Applied Sciences. It was built in a flexible fashion so that the respondents could only see the questions considered relevant based on their previous answers. The survey was anonymous and strictly voluntary; it was promoted through a mailing list and on social networks, and no reward was promised. Based on the survey, the popularity of mobile devices and the student services among the respondents as well as the shares of mobile operating systems were calculated. The descriptions of common problems and suggestions on the applications to be developed were also collected.

Surveys have proven to be an effective method of getting statistical information from a group of people. However, the information obtained through surveys is not always the most reliable (Pritibhushan 2000, 316). In this study, the main target group of the survey was new students, who are going to be the future users of the software developed with OutSystems. Webropol online survey tool was used for gathering information from the students and teachers of Lahti University of Applied Sciences. In addition, it was useful to conduct a more detailed interview with chosen students to find out their specific needs and thoughts concerning the new software.

2.2.2 Predictive extrapolation

Based on the data published by The Finnish Communications Regulatory Authority (FICORA), an assumption about how much mobile devices are used in Finland was made. Given that the data has been published bi-annually since 2007, accurate prediction of future values can be made. Based on this prediction, the conclusion about relevance of this study was made.

Predictive extrapolation is a method of creating forecasts for the future dynamics of data based on the data change history in the past. Very accurate predictions cannot be made even with big amounts of past data, but general vectors of development can be guessed (Armstrong 1984, 53). The data obtained through the above-mentioned surveys was used, as well as the data from surveys conducted in previous years. In addition to FICORA, the other source for statistical data used in this thesis is Statistics Finland, a Finnish public authority collecting statistics in Finland. This source provides much more abstract data but with much larger samples, resulting in more accurate predictions.

2.2.3 Case study – development of two applications

The case study presented in this thesis introduces OutSystems to a potential developer. During this case study, a mobile and web application were created using OutSystems, and the experience of making them was described from the point of view of a developer. As a result, a description of the advantages and disadvantages of the product was created. One of the applications reimplements Menus & Lunchtimes student service, utilizing the REST API provided by Jamix and is available online (web-based solution); the other application shows potential design improvements for Lukkarikone student service and is available for Android and iOS (mobile solution).

Several interviews were conducted with professional developers who could potentially develop applications using OutSystems, but previously

unexposed to it. During the interviews, the developers tried creating an application with OutSystems for the first time and shared their thoughts on the platform.

Based on the data collected during the case study, the thesis presents conclusions of what kinds of applications can be developed by using OutSystems and how OutSystems can answer to the problems described by the students in the survey.

2.3 The study in short

The applications reviewed in this paper utilize the client-server model, and the study focuses only on the client side of the model. OutSystems operates in a similar fashion by having a client (called Development Environment) and a server (called Platform Server), and the study mainly focuses on the client, assuming that the server is already provided.

The data collection process of this study may be repeated in the future to compare the results and examine the data dynamics, or it can be conducted in a different organizations too, not limited to universities. This study is inductive, meaning that observations are made first and then based on them, certain conclusions are drawn.

The paper contains a list of key terms and abbreviations in alphabetical order before the first chapter which can be referred to in case some specific term in the text is uncertain. The paper is divided into chapters that follow the order the research steps were taken.

3 LITERATURE REVIEW

In this study, the terms personal computers and mobile devices are used in opposition to each other. Generally speaking mobile devices discussed in this study are a subset of personal computers because smartphones and tablets are personal computers and laptops are mobile devices. However, further in this study the category of personal computers includes only desktop computers and laptops, and the category of mobile devices includes only smartphones and tablets. Some researchers include laptops, smartphones, tablets and wearable computers to the category of mobile devices, whereas others only include smartphones, excluding laptops, tablets and wearable computers.

In this study, the term "mobile devices" refers only to smartphones and tablets, and the term "personal computers" refers only to desktop computers and laptops, even though technically they are all personal computers and excluding desktop computers are all mobile devices. To make the categorizing process easier, a device is categorized according to the operating system it is running. Computers that are running Windows, macOS or any flavor of GNU/Linux are treated as personal computers and computers running Android, iOS and Windows Phone are treated as mobile devices. Computers running any other operating systems are ignored in this study because they do not represent a significant share.

3.1 Mobile applications vs. Web applications

A mobile application is a computer program developed specifically for mobile OS, and a web application is a computer program developed to run on any device inside a compatible web browser with support for certain features. The two most popular mobile operating systems are Android (45.41% of mobile devices in Europe) and iOS (38.39% of mobile devices in Europe), and the two runner-ups are Blackberry (4.59% of mobile devices in Europe) and Windows Phone (2.58% of mobile devices in Europe). The outsider is Symbian (2.32% of mobile devices in Europe),

and there are other mobile operating systems that are less popular and not counted in this study. HTML5 is commonly used for web application development and the majority of mobile devices support it with their built-in web browser. Initially, web applications seem as a better choice, but there are several factors to consider, because both types of applications have their own advantages and disadvantages. (Rowles 2014, 44, 54, 117, 119.)

3.2 Student services of Lahti University of Applied Sciences

The students of Lahti University of Applied Sciences use student services regularly on personal computers and mobile devices. The most popular student services are Reppu, WinhaWille, Lukkarikone and Menus & Lunchtimes. This study focuses the latter two. Lukkarikone replaced Reppu, a system with similar purpose and functions but inferior design, in 2015 and is being actively used by the students of Lahti University of Applied Sciences since its introduction (Ahonen & Lieskallio 2017). Menus & Lunchtimes is used to check menus at student canteens as well food ingredients and nutrition facts. The service is available as a web application as well as a mobile application for Android, iOS and Windows Phone (Salpaus 2017).

3.3 Previous survey

There was a survey conducted on BYOD initiative awareness in 2015 among the students of Lahti University of Applied Sciences. Although the topic is only vaguely related to this study, there are several relevant questions and a number of 429 respondents across 4 faculties, more than 8% of the number of the students of the university (LAMK 2017b). The answers about having personal mobile devices and the types of devices used are relevant and can be compared to the answers collected during this study. It is not possible to make accurate predictions for the future based on only two surveys, but it is possible to see the dynamics of how the number of the students using mobile devices changes.

3.4 Trends in mobile device usage

There is a general trend of shifting from desktop computers to mobile devices. According to a recent study conducted by StatCounter, prior to March 2017 Windows was the most used operating system in the world, but now Android have overtaken it for the first time (Simpson 2017). Based on this study, it can be calculated that the market share of both Android and iOS now surpasses the market share of Windows by more than 13% and both Windows and macOS combined by almost 8%.

The number of mobile devices is increasing in Finland based on the statistics of mobile data transmission volume (FICORA 2017). This study does not show the exact number of mobile devices, only the amount of mobile Internet traffic, but it can be assumed that the number of devices also grows with the amount of traffic.

3.5 Popularity of mobile devices in Finland

The Finnish Communications Regulatory Authority collects data on the volume of data transfer in mobile networks and publishes it twice a year on the Statistics Finland website. The volume of data is measured in terabytes and includes both the upload and download traffic. Based on this information, it is possible to estimate how much the population of Finland are using their mobile devices and given that the information has been published bi-annually since 2007, it is possible to predict how much people will be using their mobile devices in the future. (FICORA 2017.)

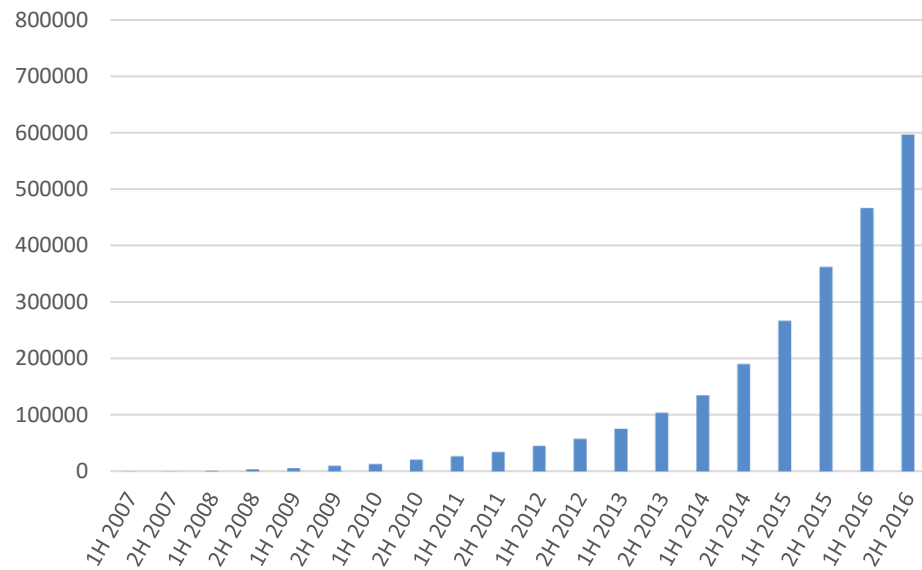


Figure 1: Consumption of Mobile Data (in Terabytes) in Finland

The statistics published by FICORA show nearly exponential growth of data transfer in mobile networks, meaning that mobile devices are used more and more every year in Finland. Having such a long history of observations, it is possible to make assumptions about the future growth of mobile data consumption and the use of mobile devices in general and expect the forecast to be more or less accurate.

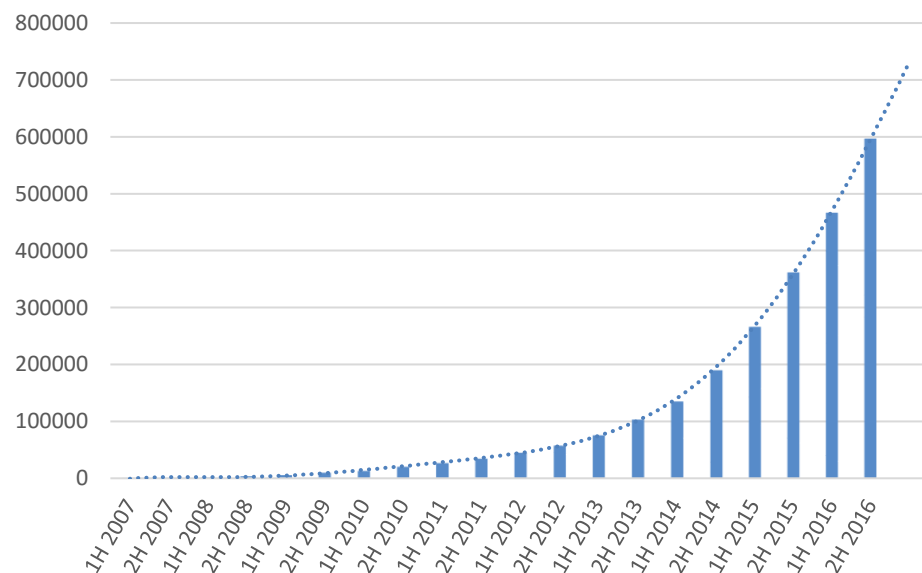


Figure 2: Forecast of the Growth of Consumption of Mobile Data (in Terabytes) in Finland

It can be expected that the volume of traffic in mobile networks will increase even more with the adoption of 5G networks, and with the research being conducted in University of Oulu, Finland is expected to be one of the first countries to introduce 5G networks (Elisa 2017). On the other hand, it is speculated that the launch of 5G networks may be delayed and thus the statistics may be unaffected by it until 2020 (Roberts 2016).

3.6 OutSystems development platform

OutSystems is a visual development environment for creating mobile and web applications. It is advertised as “the #1 low-code platform for digital transformation” and features a lot of built-in options intended to make the work of a programmer easier. According to the official website, OutSystems is used by AXA, Bacardi Limited, Volkswagen, Honda Motor Co., Ltd., GDF Suez (now known as engie), Hewlett-Packard Company, ING Groep N.V., Intel Corporation, Liberty Seguros, Mercedes-Benz, Randstad Holding NV, Siemens AG, Charles River Laboratories, Inc., Vodafone Group and many other companies across 22 industries. As for educational institutions, Bob Jones University, Georgia Institute of Technology, Kent State University and University of Georgia are already using this software. OutSystems is a Platform as a Service (PaaS) product that runs in the cloud, so the system requirements for the hardware of the computers of the developers are low, but constant high speed low latency Internet connection is required. (OutSystems 2017e.)

Platform as a Service is about to enter platform wars, and SAP Business ByDesign and Salesforce will be competing to dominate a certain segment of the market. It is still not certain if it is to be a so-called “winner takes all” market (like Microsoft among desktop operating systems) or a “winner takes most” market (like Google among search engines), and developers might be willing to switch from one platform to another easily. The biggest factor of success for a company on the PaaS market would be not the product itself or the technology behind it, but the strategy and the

ecosystem to back it up, so the most successful company would tie the developers up to their platform and prevent them from switching to competitive solutions. (Giessmann & Stanoevska-Slabeva 2012, 47.)

3.7 Rapid Application Development

Development of a mobile application is a long process which includes a lot of steps such as planning, designing, development, testing and marketing. Most of these steps require programming skills and a considerable amount of technical knowledge for each deployment platform (Rowles 2014, 119-121). It is not the case with OutSystems, which was twice named the leader in mobile low-code development by Forrester (Richardson & Rymer 2016, 9; Hammond 2017, 8). Following the principles of Rapid Application Development with OutSystems, it is possible to create a fully featured application in a very short time with almost no background knowledge or understanding of how mobile and web technologies work, which is very important for citizen developers. Because an application developed with a low-code platform contains a minimum amount of program code, its operation is very easy to change. Therefore, the development stage as well as planning and designing require less time than usual. After a mobile application is successfully deployed, only the marketing, maintenance and customer support stages are required. These stages are achieved much faster than usual with a Rapid Application Development Platform such as OutSystems (Rowles 2014, 110-111).

3.8 Internal structure of OutSystems

OutSystems is a complex cloud-based low-code development platform designed with performance, scalability and high-availability in mind. A standard Web Application Server setup is supplemented by several extra services and repositories required to generate, build, package and deploy applications. Microsoft .NET stack, Oracle WebLogic and JBOSS can be used and the application source code is generated in either C# or Java, so standard CLR or JVM technology may be used instead of proprietary

runtime. OutSystems provides a desktop application called Service Studio which provides easy access to all functions of the platform such as modelling of User Interface (UI), Business Processes, Business Logic, Databases, Integration Components, SAP BAPIs, SOAP and REST Web Services, Security Rules and Scheduling activities. Integration Studio is an application for integrating external libraries, services and databases with OutSystems used by technical developers to extend OutSystems with additional functionality. Integration Studio works in conjunction with Microsoft Visual Studio for building custom .NET components and with Eclipse for building custom J2EE components (JARs). Service Studio and Integration Studio are native Windows applications, but they do not run any of the code of the platform locally. They only connect to a remote session of a PaaS platform, so the software takes up little space in local storage but requires constant high-speed Internet connection. Both Service Studio and Integration Studio can be accessed through a web browser; therefore, developers are no longer required to install any software locally and are not limited in their choice of an operating system. (OutSystems 2015, 1-5.)

3.9 Supremacy over competitors

There are several low-code and no-code mobile application development platforms on the market. OutSystems implements several features such as 1-Click Publish, TrueChange™ and Silk UI that guarantee its superiority over the competing solutions. OutSystems was named as one of the mobile application development market leaders along with IBM, Kony, Oracle, Salesforce and SAP. OutSystems is a relatively new challenger in this field, and it tends to appeal to both business units and developers, mimicking the approach of Salesforce, an already successful platform in this area. OutSystems is oriented towards power-users acting as citizen developers. On the other hand, professional developers may not like it because it may be a threat to their work. Still, it can also be used alongside traditional application development tools. (Azoff 2015, 6-8.)

A larger-scale comparison was conducted by Gartner in which 36 different mobile application development platforms vendors were compared. OutSystems was named visionary, and its model-driven approach, support for citizen developers and inclusion of a variety of APIs were highlighted as the strengths of the platform. The lack of control over the generated code, the incompatibility with traditional development methods and the lack of certain deployment options on the other hand were marked as cautions for OutSystems, because those factors may distract potential customers from the platform. A second comparison conducted more recently moved OutSystems from the visionaries to the leaders and described it as a Rapid Application Development high-productivity Application Platform as a Service. A survey conducted by Gartner among OutSystems customers revealed that “ease of use” and “functional depth and completeness” of the platform are considered as its strengths. However, some of the respondents complained about the price of the license being unreasonably high and the quality of the documentation and technical support being very poor. (Wong, Baker, Leow & Herschmann 2016; Vincent, Baker, Natis, Iijima, Driver & Dunie 2017.)

3.10 Accessibility guidelines compliance

Since young people use mobile devices more and are likely to use mobile applications, the development of mobile applications should be a priority for organizations such as educational institutions targeting younger audiences (IPA.CO.UK 2015, 5). According to IPA, elderly people do not use mobile devices actively due to the fact that applications are often poorly optimized for use by people with certain disabilities. According to a technical note released by OutSystems (OutSystems 2008), it is fairly easy to design applications that meet the accessibility conformance level A and Level AA of Web Content Accessibility Guidelines 2.0 (W3C 2008) given that the developers are already familiar with terms and concepts of OutSystems.

3.11 OutSystems used by higher education institutions

OutSystems has successfully been adopted by several universities. They use it for application development for the researchers, for the students and directly in educational process. The reviews by The University of Georgia, Kent State University and Georgia Tech published on official website of OutSystems are very positive and tell success stories about using OutSystems in these institutions. The Rapid Application Development process is very important for such institutions because they usually have to deliver prototypes in a very limited period of time, and OutSystems copes with this remarkably well. Visual low-code development is also very appealing to the students who do not have any background experience in programming, so they can develop applications even before they learn their internal structure and the underlying principles and processes. (OutSystems 2017b; 2017c; 2017d.)

4 DATA ANALYSIS

During this study, a survey was conducted and 98 students of Lahti University of Applied Sciences took part in it. Interviews with 5 potential developers were conducted. Two applications were developed during the case study.

4.1 The survey

The link to the survey was published on several Lahti University of Applied Sciences student community pages in social networks (Facebook, VK and Yammer – total 31 respondents) and sent to randomly selected 1790 student emails (however, only 1387 emails were actually delivered and only 67 recipients took part in the survey). A total of 98 answers were collected. After discarding the answers of graduated students, 96 answers were left. Mostly young people took part in the survey, more than half of the respondents were under 25 years old and the average age of a respondent was 25.48 years. The majority of the respondents were from the Faculty of Business (54%) and the Faculty of Technology (24%), only 13% from the Faculty of Social and Health Care and 9% from the Institute of Design and Fine Arts.

The survey showed that 99% of the respondents own a smartphone and more than 10% do not own a laptop or a desktop computer at all. Only 36% of the respondents own a desktop computer and 1% do not own any personal electronic devices at all. It is speculated that smartphones may soon replace all other computers completely, thus it is very important to make student services accessible on smartphones (Bonnington 2015).

The most popular mobile operating system among the respondents is Android (64% of the respondents own an Android device), the second most popular is iOS (44% of the respondents own an iOS device), only 13% of the respondents have a Windows Phone device, 3% have other operating systems on their devices and 1% of the respondents were not competent in this matter.

The most used student service is Reppu (99% of the respondents use it), the second most used service is WinhaWille (used by 90% of the respondents), the third is Lukkarikone (used by 86% of the respondents). Menus & Lunchtimes is used only by 58% of the respondents, probably because it is not essential for the studies and many students are not aware of it. Other services used by the students include Yammer, Respa, Office 365 and Mobility online.

44% of the respondents find their experience of using student services on mobile devices positive, 47% of the respondents find their experience of using student services on mobile devices negative and 9% did not happen to have such experience at all. The most common complaints about the web versions of the services are that they are not optimized for use on small resolutions, not being compatible with certain web browsers and performance issues.

57% of the respondents would like to install applications developed by Lahti University of Applied Sciences and only 13% would not. The most demanded applications for mobile devices are Reppu, Lukkarikone and WinhaWille. A popular suggestion is to create a single application that includes functions of all the student services. 27% of the respondents are willing to create their own mobile applications and most of their suggestions may easily be implemented with OutSystems, some suggestions are also very similar to student services applications, so perhaps these students may be potential application developers.

4.2 Comparison with the previous survey

The first survey was conducted in 2015 and the answers from 429 students were collected, the second survey was conducted in 2017 and the answers from 96 students were collected. It is possible to assume that both surveys represent the same social groups in similar proportions because the methods of conducting the surveys were similar and the gender, age and the faculty of studying shares are very similar. It can be

seen on the following bar charts where study from 2015 is shown in blue and the study from 2017 is shown in red.

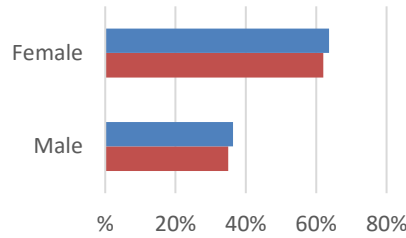


Figure 3: Gender share of the respondents in 2015 (blue) and 2017 (red)

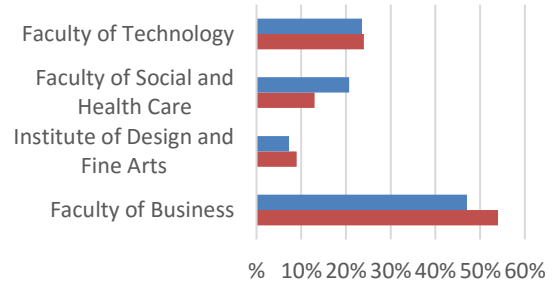


Figure 4: Faculty share of the respondents in 2015 (blue) and 2017 (red)

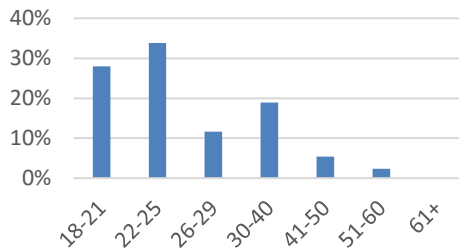


Figure 5: Age share of the respondents in 2015

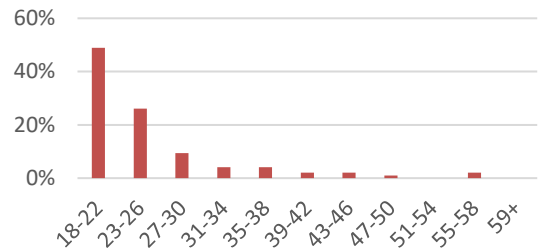


Figure 6: Age share of the respondents in 2017

The differences are that since 2015 the share of the respondents not owning a personal electronic device decreased from 7% to just 1%, the share of the respondents having a smartphone increased from 58% to 99% and the shares of the respondents owning tablets and desktop computers also increased by 23% and 21% respectively, however the share of the laptop owners among the respondents remains almost the same.

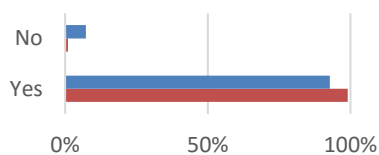


Figure 7: Share of the respondents admitting having a personal electronic device in 2015 (blue) and 2017 (red)

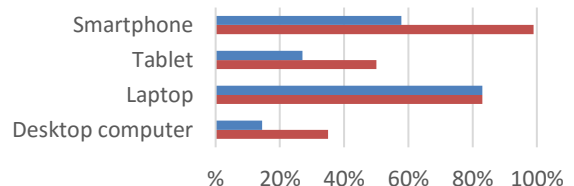


Figure 8: Shares of the respondents for each type of personal electronic devices in 2015 (blue) and 2017 (red)

It is possible to compare the results of the two surveys conducted among the students of Lahti University of Applied Sciences, but it is not reasonable to make any forecasts based only on only two surveys, so only the changes in the statistical data were noted.

4.3 The interviews

A total of five interviews was conducted and three out of five developers showed their interest in OutSystems, the other two stated that they were not interested. During the interviews, the developers previously unexposed to OutSystems tried to develop an application using it. The interviews were conducted in informal atmosphere and had a flexible structure of a casual conversation in order to make the interviewees feel more comfortable. The interviews were recorded and additional written notes with time marks were taken in order to make the analysis of the recordings easier.

The interviewees were asked to try to create a simple application using OutSystems without reading the documentation and all five of them did succeed, however all of them had troubles identifying control elements because they are not always clearly distinguishable. All the developers interviewed noticed that the overall layout of OutSystems development environment is very similar to Microsoft Visual Studio, IntelliJ IDEA, Eclipse and other popular integrated development environments, so common interface elements were already in the places that professional developers are used to have them in. Most of the interviewees were impressed by the speed of code generation and the convenience of autonaming variables.

All five interviewed people found both advantages and disadvantages of the development platform. Most advantages were related to the ease of development and deployment, fast application delivery and innovativeness of visual representation of traditionally text-based structures. Two out of five developers stated that OutSystems may be suitable for people who are learning programming, because it provides evident visualizations of

the algorithms of computer programs. One of the interviewees complemented the uniform look of a platform and inability of the environment to be customized by its users.

On the other hand all five developers criticized the aspects related to the Platform as a Service nature of the development environment like requiring constant high-speed and low-latency Internet connection, fully depending on the remote server, not sharing common clipboard with the rest of the system and constantly losing keyboard focus so that the development environment should be reloaded once in a while. Only one of the interviewees had previous experience with cloud-based IDEs and the other four discovered these problems for the first time.

5 THE CASE STUDY

In addition to the conducted interviews a web application and a mobile application were developed within the case study.


5.1 OutSystems as a Development Platform overview

During this case study, OutSystems development environment was used by the author of this paper to develop several applications. The author was not a professional developer, but had previous programming experience and understanding of visual Rapid Application Development based on previous work with ThingWorx, an IoT development platform adopted by Elisa.

5.1.1 Steps to create an application with OutSystems

Given that OutSystems provides visual development environment, it is more convenient to illustrate each step of creating an application with a picture. The pictures are provided only for reference and any of the functions described and any of the interface elements shown in this chapter may be changed or removed by OutSystems in the future.

If the OutSystems development environment was installed correctly, a “Connect to Environment” button should be visible.



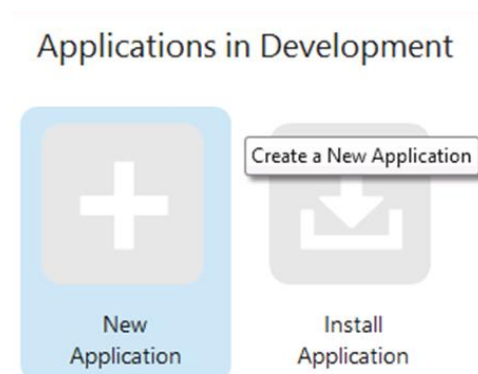
CONNECT TO ENVIRONMENT

Picture 1: The “Connect to Environment” button

In order to connect successfully, the Internet connection should be stable, fast-speed and low-latency. If the internet connection does not satisfy these requirements, freezes, interrupts and potential loss of data may be expected. If a proxy is used to access the internet, proxy authentication details should be entered in advance in the Preferences.

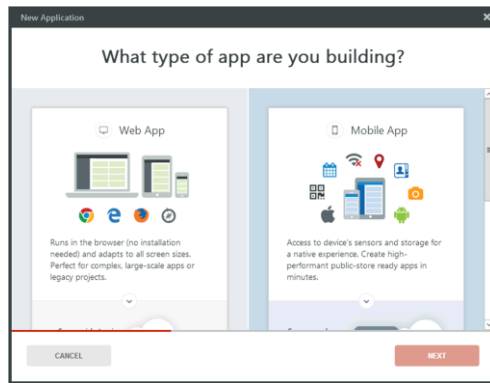
Picture 2: The credentials entry window

After the “Connect to Environment” button is clicked, the credentials entry window should appear. The user is asked to enter the URL of a server to connect to, the username (email) and the password, then to click the “Connect” button. After successful connection this window should disappear.

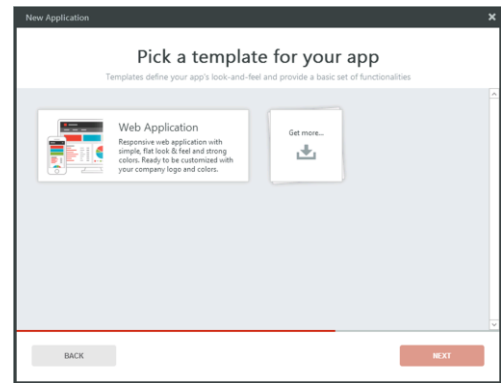


Picture 3: The “New Application” button

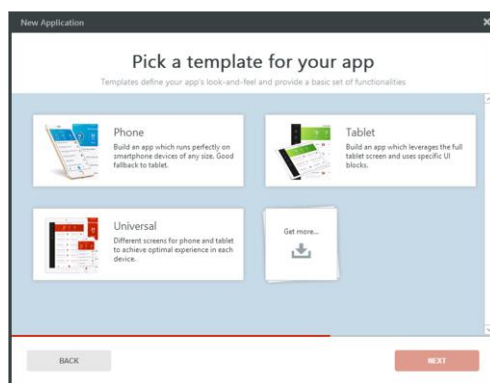
The screen containing icons for every application of current user on selected server should appear. To create a new application, the user should click the “New Application” button. After this the application creation wizard window should appear and the user is asked to select the type of the application to be developed, a template to start with and a name and base color for the application. Optionally the user can enter a description for the application and upload an icon to replace the one that is generated automatically based on the name and base color that were chosen.



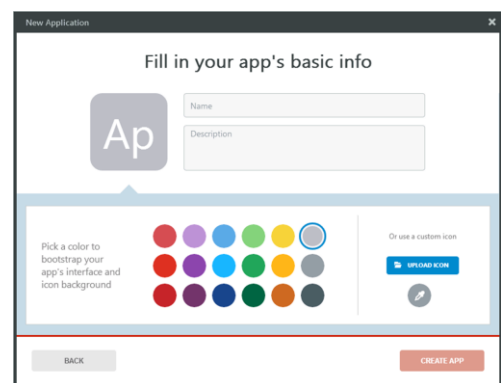
Picture 4: Choosing the type of the future application



Picture 5: Available template if web application was selected



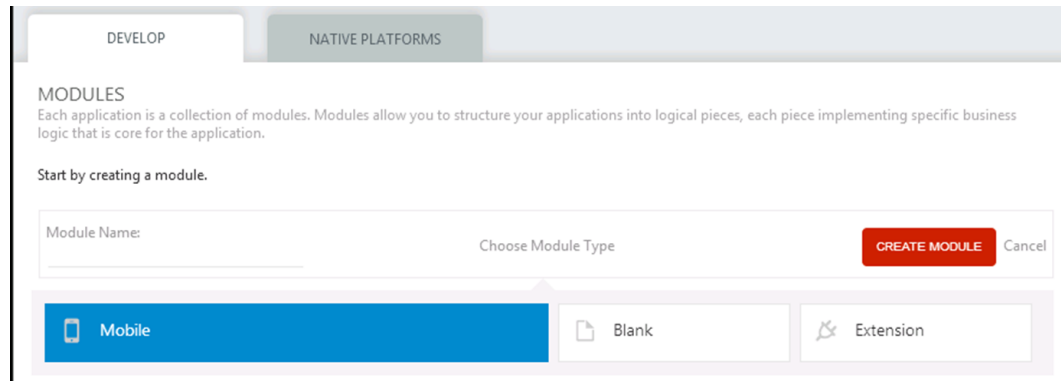
Picture 6: Available templates if mobile application was selected



Picture 7: Entering the name and picking base color for the future application

The steps of creating an application using the application creation wizard are shown on the pictures above. Currently there is only one default template for Web Applications and three different default templates for Mobile Applications targeting phones, tablets and both accordingly.

After an application is created, at least one module should be created in order to start the development process. A "Create Module" button should be visible and all necessary information for module creation should be already filled in with names automatically generated based on previous input.

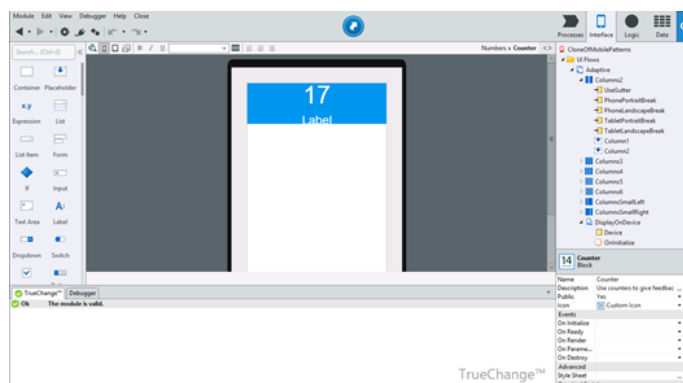


Picture 8: Creating the first module of the application

After creating the module the OutSystems enters the module edit mode. It is possible to switch between different modules and applications by going back to the main screen from the module edit mode, selecting a module and editing it.

5.1.2 Development environment

After entering the module edit mode the development environment should change to distinctive IDE interface. The developers interviewed compared the graphical user interface of OutSystems development environment to Microsoft Visual Studio, IntelliJ IDEA, Eclipse and other professional IDEs. If the developer is already familiar with at least one of these products, it should be comfortable to develop with OutSystems.



Picture 9: Typical screen of OutSystems development environment

The layout of the screen is built to be intuitive for a developer and is aligned to a flexible grid. The size of the elements of the grid can be

changed, however the elements cannot be switched places in order to keep the consistency of the development environment. The top bar contains the menu, navigation and action pictograms, the “1-Click Publish” button and a ribbon for switching tabs of a tree element on the right.



Picture 10: “1-Click Publish” green button

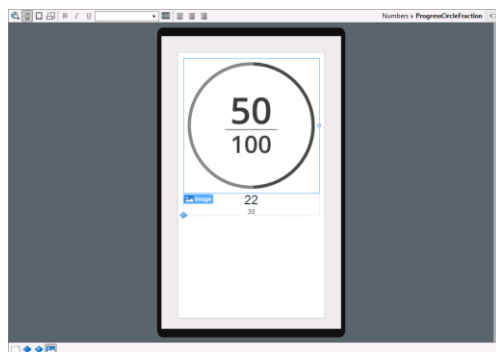


Picture 11: “1-Click Publish” red button

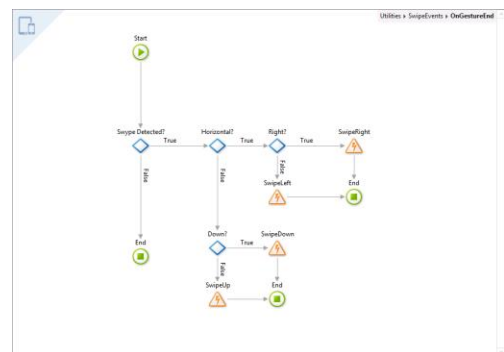


Picture 12: “Open in Browser” blue button

The biggest and the most important element of the interface is the “1-Click Publish” button which has two functions: performing an action when clicked and indicating the application state and also the action that is performed when clicked. The three states are shown on a picture above: “1-Click Publish” (green) means that the application contains changes that are valid and ready to be deployed, the deployment starts when clicked; “1-Click Publish” (red) means that the application contains changes that are invalid and deployed is impossible, the error message is highlighted when clicked; “Open in Browser” (blue) means that the application does not contain any changes compared to the previously published version, the application is opened when clicked.



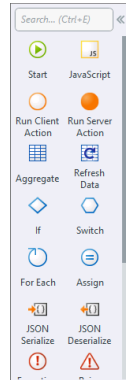
Picture 13: Visual interface design editor



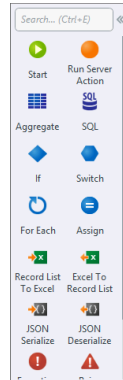
Picture 14: Visual logic design editor

As opposed to traditional text-based IDEs the largest area of the screen is occupied by a visual design editor. The elements of the design editor can be dragged with mouse gestures and manipulated with only the cursor movements. This makes the use of keyboard in this IDE optional, which is not possible in traditional development which involves a lot of typing. The

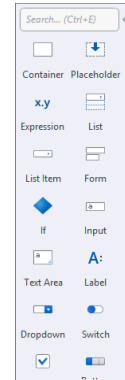
visual design editor can change to logic design mode and interface design mode if a logic or interface element was double-clicked in the tree accordingly.



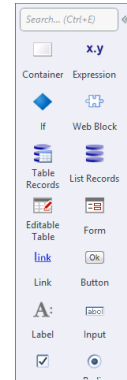
Picture 15: The set of icons for a Client Action



Picture 16: The set of icons for a Server Action

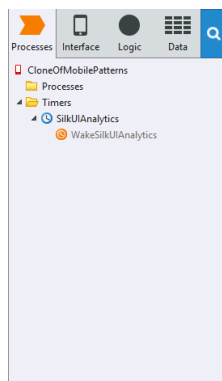


Picture 17: The set of icons for a Mobile interface design

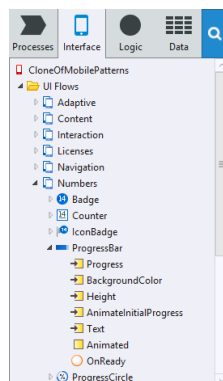


Picture 18: The set of icons for a Web interface design

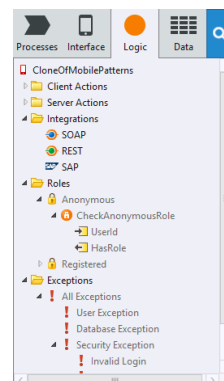
The icons on the left panel also change accordingly to what is shown in the visual design editor. The icons representing logic elements are shown along with visual logic design editor and the set of icons depends on whether a client action or server action is selected; the icons representing widgets are shown along with visual interface design editor and the set of icons depends on the type of the UI of the application.



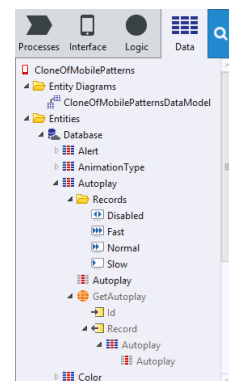
Picture 19: Process tree tab



Picture 20: Interface tree tab



Picture 21: Logic tree tab



Picture 22: Data tree tab

On the right side the visual design editor the trees of the elements of the application are situated. There are four tabs entitled “Processes”, “Interface”, “Logic” and “Data” and each of them has a separate tree behind it. These four trees represent the structure of the application and

provide easy access to all of its elements in a categorized manner. One of the key features of OutSystems is adding external data through “Integrations” category in logic tree. Currently it supports SOAP (using XML data), REST (using JSON data) and SAP (using BAPI data). In the top-right corner there is a search button used to easily find the element, given that the name of the element is known, but the category is not.

ProgressCircle Block	
Name	ProgressCircle
Description	Circular progress indicator, w ...
Public	Yes
Icon	Custom Icon
Events	
On Initialize	
On Ready	OnReady
On Render	
On Parameters Changed	OnParametersChanged
On Destroy	
Advanced	

Picture 23: The table of properties of a block

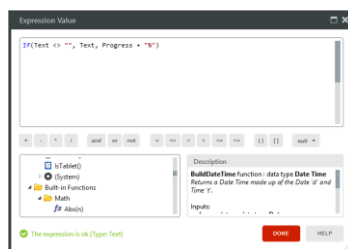
OfflineDataSync Script	
Name	OfflineDataSync
Description	
Public	Yes
JavaScript	(function) {
Runtime Path	/CloneOfMobilePatterns/scripts/C
Required Scripts	
Add Required Script	

Picture 24: The table of properties of a script

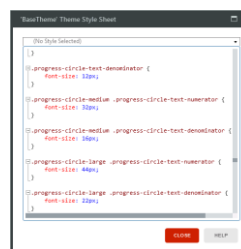
CustomLicenses	
Name	CustomLicenses
Description	Add your custom licenses he ...
Align	
Layout	
Width	(fill parent)
Min. Height	
Margin Top	Auto
Margin Left	
Style Classes	"ph"
Attributes	
Property	
Value	

Picture 25: The table of properties of an element

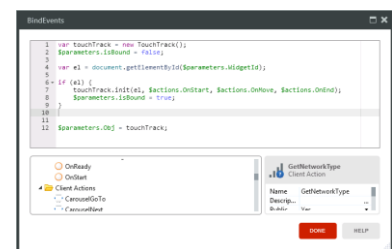
Under the tree of the elements the table of properties is situated. It shows all the properties of the select element and allow to change them. If the property is invalid, a red outline is displayed around the cell containing its value. There a lot of different possibilities regarding the tables of properties, so only three distinctive samples are provided.



Picture 26: OutSystems internal code editor



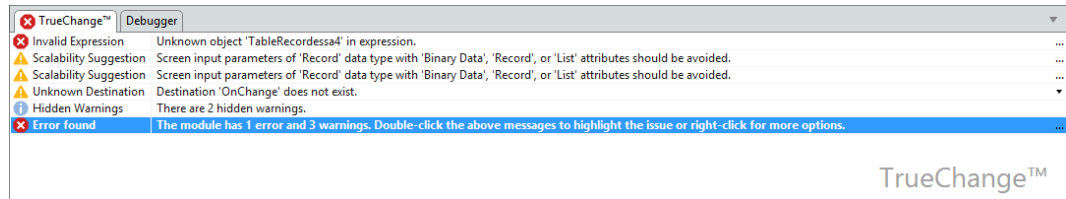
Picture 27: CSS editor



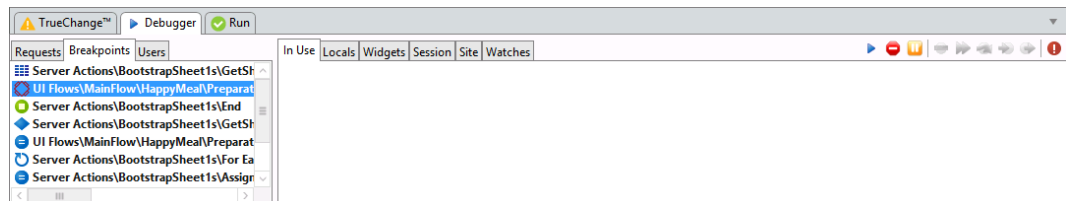
Picture 28: JavaScript editor

OutSystems is a low-code platform, so for some tasks little amounts of code should be written using internal OutSystems programming language. It is very simple and contains a lot of built-in functions that can be drag-and-dropped into the text entry field. It is advised to use agile methodologies, in particular RAD for application development, although SCRUM is used during the development process of OutSystems (Cabot

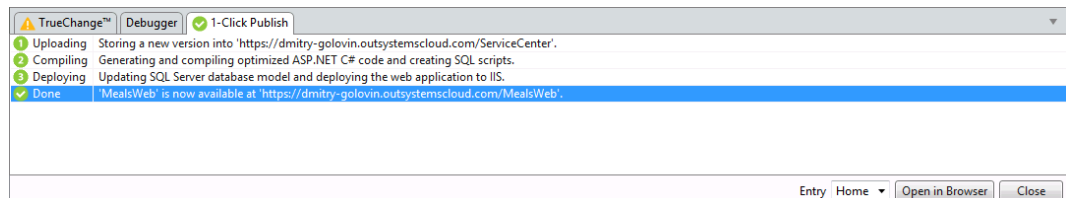
2016). For styling of the application CSS editor is available and for more advanced client-side code JavaScript editor provides a superset of ECMA-262 standard with multiple useful built-in functions. This combination of three languages provides a very convenient application development using the latest web standards. Syntax highlighting function is provided for all three editors.



Picture 29: TrueChange™ detected that the application has 3 warnings and 1 error



Picture 30: Debugger tab showing multiple breakpoints in the application

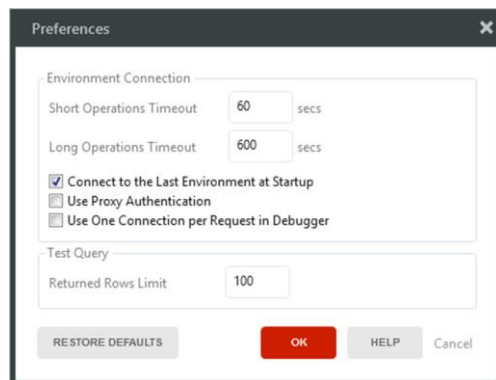


Picture 31: 1-Click Publish tab output after successful deployment of the application

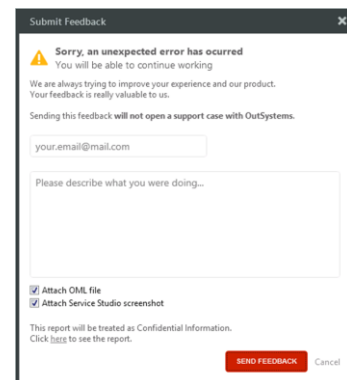
The element on the bottom of the development environment is usually hidden and only shown when being used. By default it has only two tabs “TrueChange™” and “Debugger”, but more different tabs may be added automatically if they are needed. It is used for the IDE to communicate with the users and point them to certain places in the application. With this kind of intelligent development environment bi-directional communication between a human and a computer program is essential part of successfully development of an application.

5.1.3 Customizations and handling crashes

OutSystems development environment provides the minimum amount of settings, so that the look of it cannot be customized. This has the advantage of the consistency of the environments for different developers, so it differs only slightly on different computers with different resolutions. The main advantage of the uniform look is the ease of creating instructional videos and technical support. All the preferences are limited to the ones that are related to the cloud connectivity. If the development environment tries to perform an invalid operation, it crashes and suggests the users to submit their feedback, but not to open a support case.



Picture 32: Preferences window



Picture 33: Crash feedback window

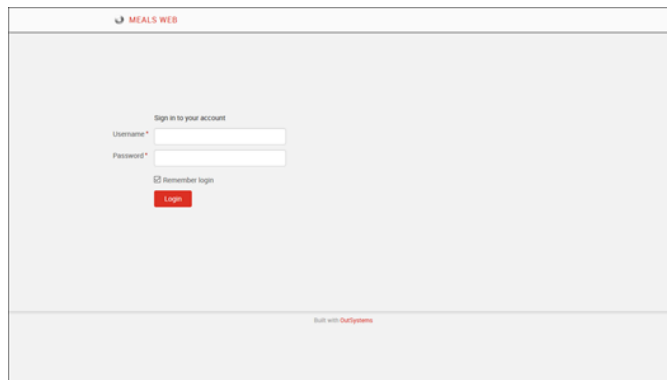
On the other hand this stable consistency may be broken with an update that is installed without notifying the user. At the moment there are a lot of consistency and stability problems with OutSystems, but it can be fixed remotely with future updates in virtue of Platform as a Service nature of the development environment.

5.2 Alternative web application for Menus & Lunchtimes

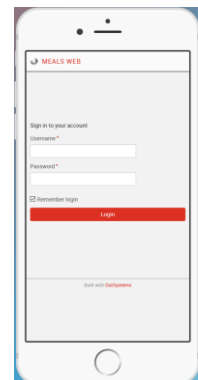
Menus & Lunchtimes is an application used by the students of Lahti University of Applied Sciences to check the ration of the student canteen, its ingredients and nutrition facts. There are four different applications called Jamix Menu for this: a web application, an Android application, an iOS application and a Windows Phone application. Although those

applications exist, most of the respondents do not use them or even not aware of them. (Salpaus 2017.)

During this case study, a web application was developed using OutSystems, so that all students can access OutSystems application on personal computers as well as on mobile devices. For the reference screenshots of a personal computer interface the resolution of 1366x768 pixels was chosen as it happens to be the most common laptop display resolution for historical reasons (Remaker 2017). The reference screenshots of a mobile device interface are made using a simulator of an Apple iPhone 6 Plus with a resolution of 414x736 on a 5.5-inch screen giving the pixel density of 401 pixels per inch.

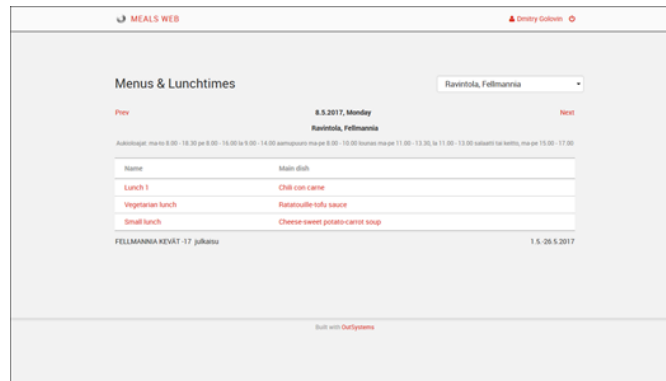


Picture 34: The credentials entry page as seen on a personal computer



Picture 35: The same page as seen on a mobile device

When using this application the user is first asked to log into the system. This mechanism is automatically generated by OutSystems and can be used to remember the last canteen the user selected to go back to it the next time the application is accessed, save information about food allergies of the user (like lactose or gluten intolerance) to warn user about food containing hazardous ingredients, as well as food preferences to notify the user about favorite dishes being served in a nearby canteen. The existing application does not provide such functions.

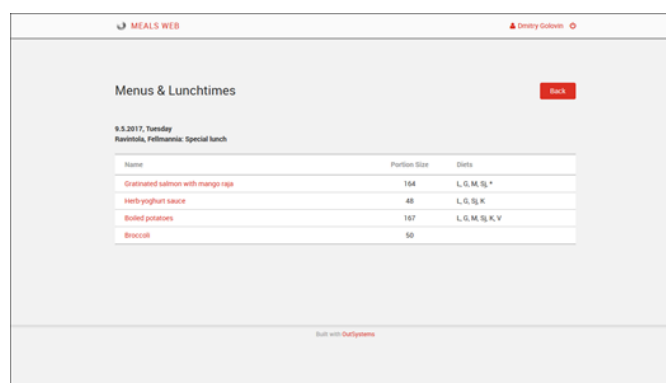


Picture 36: The menu browsing page as seen on a personal computer

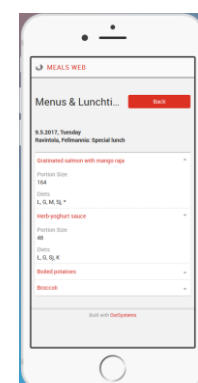


Picture 37: The same page as seen on a mobile device

The first page of the application that the user can see is the list of meals being served today in the last canteen he previously browsed. The canteen can be selected from a drop-down list in the top-right corner, the date may be changed by using the “Prev” and “Next” links to go to yesterday or tomorrow accordingly relatively to the current date shown in between these two links. Current canteen name, brief information about it like opening hours and current menu name is also displayed. Mobile version has similar style, but rendered in a slightly different fashion: the table is truncated, so the user has to tap on an arrow in order to display the main dish of a meal.



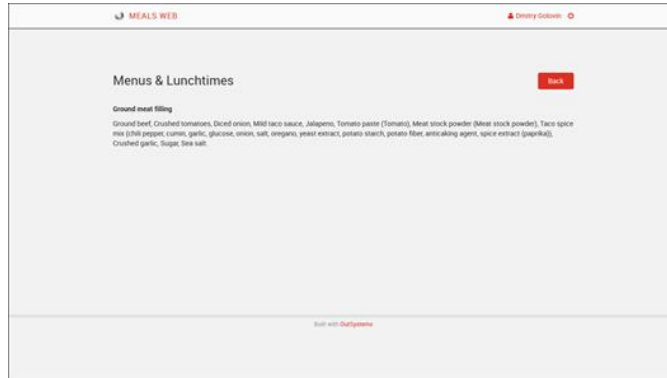
Picture 38: The menu details page as seen on a personal computer



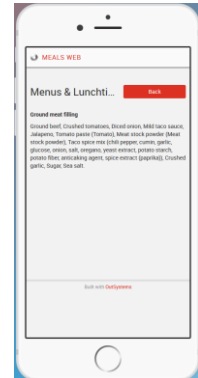
Picture 39: The same page as seen on a mobile device

Once the user clicks (desktop version) or taps (mobile version) on the name of the meal or the main dish, he is sent to the details page, where all the dishes of the meal are displayed along with the information about portion size and diet warnings. Mobile version has similar behavior of the

arrows as on the first page. The user can go back to the first page by clicking or tapping the “Back” button or see the ingredients of each dish by clicking or tapping on its name.



Picture 40: The ingredients page as seen on a personal computer



Picture 41: The same page as seen on a mobile device

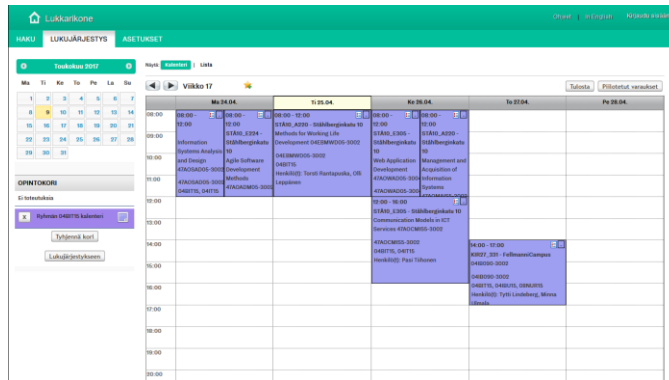
On the ingredients page there are no differences between desktop and mobile versions, except that the text is formatted differently due to adaptive design. The user can go back by clicking or tapping the “Back” button.

5.3 Android and iOS application design for Lukkarikone

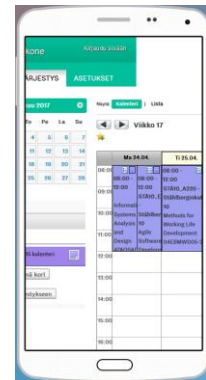
The survey about Mobile and Web applications showed that the respondents often face problems when using Lukkarikone on mobile devices, because the application is trying to output too much information for a limited screen area and being not optimized for touchscreen input. Given that this application was introduced only recently and the students are mostly satisfied with how it works on a desktop computer, it should not be replaced or modified and alternative mobile application should be developed instead to coexist with web version for personal computers.

Most of the students who was asked about their experience with this service stated that Lukkarikone is usable on desktop computer, but almost impossible to use on mobile device. To illustrate it the reference screenshots of a personal computer rendering with the resolution of

1366x768 pixels and Samsung Galaxy S4 simulation with real resolution of 1080x1920 pixels and CSS resolution of 360x640 are provided.

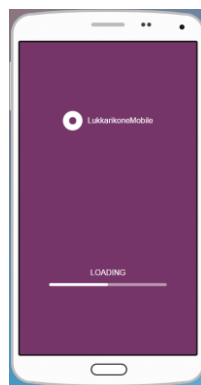


Picture 42: Existing Lukkarikone application as seen on a personal computer

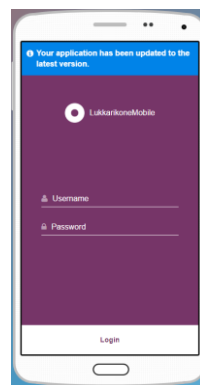


Picture 43: The same application as seen on a mobile device

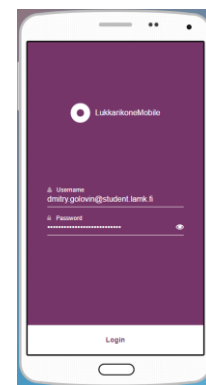
During this case study, an alternative mobile application design for Lukkarikone was developed. The prototype was called Lukkarikone Mobile and tested on the same simulation of a Samsung Galaxy S4. Most of the application logic and design was automatically generated by OutSystems allowing the developer to focus on the design and logic of the important parts.



Picture 44: Loading screen



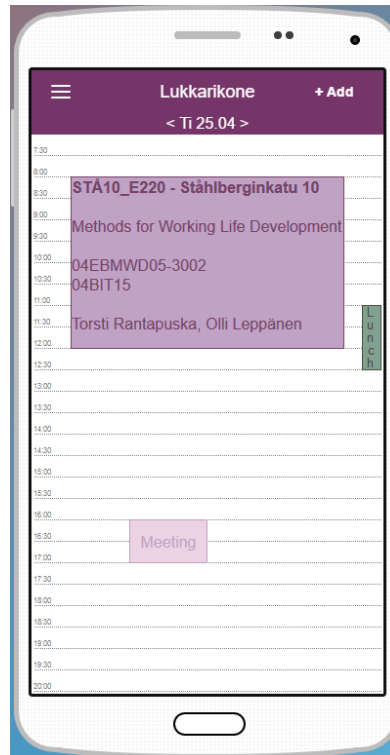
Picture 45: Automatic update notification



Picture 46: Entering credentials to log in

OutSystems provides the design and automatically generated code for the loading screen, automatic updates, secure account management and many more so the developer does not have to implement these common

features. It is possible to change any of these components if the usage of default components is not acceptable for some reason.



Picture 47: Lukkarikone Mobile interface suggestion

Similar to the Menus & Lunchtimes application, Lukkarikone Mobile is showing much less information at the same time on the screen of a mobile device compared to the version targeted at desktop computers. The user can customize the colors of the events in the calendar and save color preferences, custom events can be added. These functions are not currently available in the existing application. Lukkarikone Mobile provides time travel features similar to Menus & Lunchtimes, but instead of using the “Prev” and “Next” links, specific touch gestures could be used, because the application is only targeted at mobile devices equipped with a touch screens, so the users can swipe left to go to yesterday and right to go to tomorrow relatively to the current date shown in the center. The other touch gestures used are scroll to access invisible parts of the table, pinch in to increase the scale and see more details, spread out to decrease the scale and go to week or month overview and slide to access the menu.

6 CONCLUSIONS

Several different research methods were used during this study, and the combination of the outcomes of the study can provide a good basis for making specific conclusions.

Based on the extrapolation of the data from FICORA, it is possible to assume that people in Finland will be using more and more mobile devices in the next few years, and the shift from desktop computers to mobile devices will be gaining further momentum. All this makes this study relevant and guarantees that it will not become useless in the foreseeable future.

Based on the survey, the major needs and complaints of the students of Lahti University of Applied Sciences were determined, so the priorities for application development may be created. The most used student services are Reppu (99%), WinhaWille (90%) and Lukkarikone (86%). Most of the complaints about incompatibility on mobile devices concern WinhaWille and Lukkarikone, so their application development should be a priority. The design proposition for Lukkarikone Mobile is made in this paper. The idea of creating a single extensible meta-application that will have the functions of all existing student services and even more was noted by many respondents, so this should also be considered if the cost-benefit ratio of development of such an application can be expected to be reasonable.

The case study reveals how OutSystems can be used to develop mobile and web applications. Most of the interviewed professional developers admitted that OutSystems is a superior Rapid Application Development platform and expressed their desire to use it. Some of the respondents of the survey also expressed similar desire, and their suggested application ideas resemble the needed applications. Therefore, it would be advisable to consider the students as potential citizen developers. The case study also contains basic instructions on how to use OutSystems that should be a good kick start for new developers.

It can be concluded that student services can be reimplemented using OutSystems in such fashion that the majority of the students can be satisfied. The three services that have the highest priority to have a mobile device-friendly interface are Reppu, WinhaWille and Lukkarikone. Since there already is a Reppu mobile application, it should not be reimplemented. However, its problems should be fixed, and students should be made better aware of the existence of the application.

Considering that OutSystems has been successfully used by many businesses for a long time and the ease of adopting and using it, Lahti University of Applied Sciences can also take advantage of it. The students without programming experience will be able to create mobile and web applications as citizen developers, and students experienced with C# or Java will be able to produce certain modules specific to the university and integrate them with OutSystems. Fair productivity of student application development teams may be achieved by utilizing the principles of Rapid Application Development. The mobile applications produced by OutSystems are compatible with Android- and iOS-based devices which majority of the students and teachers use, so the applications developed by the students for Lahti University of Applied Sciences can be used by almost anyone in the university.

7 SUMMARY

The purpose of this study is to determine the main problems that the students of Lahti University of Applied Sciences experience and find out how these problems can be solved with OutSystems that is already used and appreciated by many companies and other organizations. The concepts behind the Platform as a Service are rather old and are described by a number of researchers.

During the study, a survey and five interviews with randomly selected professional developers were conducted. Based on the survey in which 98 students of Lahti University of Applied Sciences took part, the satisfaction level and the most important problems were identified. Based on the interviews, the strengths and the weaknesses of OutSystems as a Rapid Application Development Platform for Mobile and Web were determined and described.

During the case study, a web application and a mobile application were developed. The web application shows an example integration with third-party service and a flexible and responsive mobile friendly design that is rendered differently on desktop computers and mobile devices so that it is automatically adjusted to present information in a most comprehensible way. The mobile application shows a conceptual design of an existing student service adapted to be used in a mobile application with mobile-specific features utilized for navigation within the application. The design solutions presented are not limited to these applications and may be used elsewhere. An easy to understand guide to start the development from ground-up was written that can be used as an instructional manual for the new developers.

Based on the conducted research, the conclusions about the preferred order and the optimal methods of development of the mobile applications for the student services were made. The suggestions about the design from the case study should also be taken into consideration in order to build an application to be accepted by the majority of the users.

LIST OF REFERENCES

- Ahonen, A. & Lieskallio, M. 2017. Lukujärjestykset - Timetables. LAMK wiki [accessed 8 May 2017]. Available at: <https://wiki.lamk.fi/pages/viewpage.action?pageId=37552727>
- Armstrong, J.S. 1984. Forecasting by Extrapolation: Conclusions from Twenty-five Years of Research. *Interfaces*, November 1984, Volume 14, Issue 6. Pages 52-66.
- Azoff, M., 2015. Ovum Decision Matrix: Selecting a Mobile App Development Platform Solution, 2015–16. Ovum. Available at: <https://www.ovum.com/research/ovum-decision-matrix-selecting-a-mobile-app-development-platform-solution-2015-16/>
- Bonnington, C. 2015. In Less Than Two Years, a Smartphone Could Be Your Only Computer. *WIRED* [accessed 5 May 2017]. Available at: <https://www.wired.com/2015/02/smartphone-only-computer/>
- Cabot, J. 2016. Interview with OutSystems, a model-driven company that raised \$55M in funding. *Modeling Languages* [accessed 10 May 2017]. Available at: <http://modeling-languages.com/outsystems-a-model-driven-company-raises-55m-funding/>
- Davis, D., Golicic, S. & Boerstler, C. 2011. Benefits and challenges of conducting multiple methods research in marketing. *Journal of the Academy of Marketing Science (JAMS)*. June 2011, Volume 39, Issue 3. Pages 467-479.
- Elisa. 2017. Elisa promotes world class research. Elisa [accessed 5 May 2017]. Available at: <http://corporate.elisa.com/news-room/press-releases/elisa-press-release/?otsikko=elisa-promotes-world-class-research&id=94929088699237&tag=all>
- The Finnish Communications Regulatory Authority (FICORA). 2017. Mobile data transmission volume. Prime Minister's Office and Statistics Finland [accessed 29 May 2017]. Available at: <http://www.findikaattori.fi/en/125>
- Gartner IT Glossary 2017. Citizen Developer. Gartner [accessed 11 May 2017]. Available at: <http://www.gartner.com/it-glossary/citizen-developer/>
- Giessmann, A. & Stanoevska-Slabeva, K. 2012. Business Models of Platform as a Service (PaaS) Providers: Current State and Future Directions. *Journal of Information Technology Theory and Application (JITTA)*. Volume 13, Issue 4, pp. 31-55, December 2012.

Hammond, J. 2017. The Forrester Wave™: Mobile Low-Code Development Platforms, Q1 2017: The 11 Providers That Matter Most And How They Stack Up. Forrester. Available at:
<https://www.forrester.com/report/The+Forrester+Wave+Mobile+LowCode+Development+Platforms+Q1+2017/-/E-RES136055>

IPA.CO.UK. 2015. TouchPoints5 - The Power of Passive Data. IPA [accessed 1 June 2017]. Available at:
<http://www.ipa.co.uk/document/touchpoints-5--the-power-of-passive-data>

Lahden ammattikorkeakoulu (LAMK), 2017a. History. LAMK.FI [accessed 4 May 2017]. Available at:
<http://www.lamk.fi/english/about/organisation/history/Sivut/default.aspx>

Lahden ammattikorkeakoulu (LAMK), 2017b. Organisation. LAMK.FI [accessed 4 May 2017]. Available at:
<http://www.lamk.fi/english/about/organisation/Sivut/default.aspx>

Mikkonen, A. 2013. Mikkelissä on Suomen paras ammattikorkeakoulu - Talouselämä listasi Suomen parhaat amk:t. Talouselämä [accessed 4 May 2017]. Available at: <http://www.talouselama.fi/uutiset/mikkelissa-on-suomen-paras-ammattikorkeakoulu-talouselama-listasi-suomen-parhaat-amk-t-3444574>

OutSystems. 2008. TechNote — Web Content Accessibility 2.0. OutSystems [accessed 1 June 2017]. Available at:
<http://www.outsystems.com/community/download.aspx?DocumentId=123>

OutSystems. 2015. OutSystems® Platform Architecture and Infrastructure Overview. OutSystems [accessed 1 June 2017]. Available at:
<http://www.outsystems.com/goto/outsystems-platform-architecture-overview>

OutSystems. 2017a. Downloads. OutSystems [accessed 5 May 2017]. Available at: <https://www.outsystems.com/home/downloads/>

OutSystems. 2017b. Research Center Develops Lab Management Solution to Process 150,000 Reservations per Year. OutSystems [accessed 5 May 2017]. Available at: <https://www.outsystems.com/case-studies/georgia-tech/lab-management-solution/>

OutSystems. 2017c. Kent State Adopts Rapid Application Development Platform to Better Serve Students, Faculty and Administrators. OutSystems [accessed 5 May 2017]. Available at:
<https://www.outsystems.com/case-studies/kent-state/rapid-application-development-platform/>

- OutSystems. 2017d. OutSystems Empowers Educators to Better Prepare University of Georgia Graduates. OutSystems [accessed 5 May 2017]. Available at: <https://www.outsystems.com/case-studies/uga/devops-tool-for-business-and-it-majors/>
- OutSystems. 2017e. The #1 Low-Code Platform for Digital Transformation – OutSystems. OutSystems [accessed 1 June 2017]. Available at: <https://www.outsystems.com/>
- Pritibhushan, S. 2000. Determination of reliability of estimations obtained with survey research: A method of simulation. *International Journal of Market Research*. Summer 2000, Volume 42, Issue 3. Pages 311-317.
- Remaker, P. 2017. How 1366 x 768 Became A Common Laptop Screen Resolution. *Forbes via Quora* [accessed 9 May 2017]. Available at: <https://www.forbes.com/sites/quora/2017/03/10/how-1366-x-768-became-a-common-laptop-screen-resolution/>
- Richardson, C. & Rymer, J. 2016. The Forrester Wave™: Low-Code Development Platforms, Q2 2016: The 14 Providers That Matter Most And How They Stack Up. Forrester. Available at: <https://www.forrester.com/report/The+Forrester+Wave+LowCode+Development+Platforms+Q2+2016/-/E-RES117623>
- Roberts, M. 2016. 5G will hit 24 million subscriptions worldwide in 2021. *Ovum* [accessed 5 May 2017]. Available at: https://www.ovum.com/press_releases/5g-will-hit-24-million-subscriptions-worldwide-2021/
- Rowles, D. 2014. *Mobile marketing: How mobile technology is revolutionizing marketing, communications and advertising*. Kogan Page Limited.
- Salpaus. 2017. Ruokalistat. Koulutuskeskus Salpaus [accessed 8 May 2017]. Available at: <http://www.salpaus.fi/asiakaspalvelut/kookuppilat/ruokalistat/Sivut/default.aspx>
- Simpson, R. 2017. Android overtakes Windows for first time. *StatCounter* [accessed 1 June 2017]. Available at: <http://gs.statcounter.com/press/android-overtakes-windows-for-first-time>
- Vincent, P., Baker, V., Natis, Y., Iijima, K., Driver, M. & Dunie, R. 2017. *Magic Quadrant for Enterprise High-Productivity Application Platform as a Service*. Gartner. Available at: <https://www.gartner.com/doc/3695317/magic-quadrant-enterprise-highproductivity-application>

Wong, J., Baker, V., Leow, A. & Herschmann, J. 2016. Magic Quadrant for Mobile App Development Platforms. Gartner. Available at: <https://www.gartner.com/doc/3348018/magic-quadrant-mobile-app-development>

World Wide Web Consortium (W3C). 2008. Web Content Accessibility Guidelines (WCAG) 2.0. W3C [accessed 1 June 2017]. Available at: <https://www.w3.org/TR/WCAG20/>