

Patrick Wolff

Development of Customer Recognition Plugin for Contact Center Application

Helsinki Metropolia University of Applied Sciences

Master's Degree

Information Technology

Master's Thesis

19 March 2016

Author(s) Title Number of Pages Date	Patrick Wolff Development of Customer Recognition Plugin for Contact Center Application 50 pages + 1 appendices 19 March 2016
Degree	Master of Engineering
Degree Programme	Information Technology
Instructor(s)	Kari Salo, Principal Lecturer
<p>This thesis describes the development of a customer recognition plugin for adoption into a set application environment with several third party application suites not syncing together out of the box. The aim of the thesis is to give an example of how quick wins might make a big difference, but also to function as an input for future development in the areas of CRM and contact center applications.</p> <p>Within the set framework a working plugin webpage was developed, features and dependencies analyzed and the process re-designed. At the end there is also a discussion regarding future proofing the processes and examples of what could be the next steps regarding application and platform integrations.</p>	
Keywords	Contact center, recognition, integration

Contents

Abstract

Table of Contents

Abbreviations

1	Introduction	1
2	Methods and Materials	4
2.1	Context of Project	4
2.2	Research Design and Research Process	4
2.3	Planned Research Strategy	5
2.4	Overview of Planned Data Collection and Data Analysis	5
2.5	Development Methods	5
3	Background	7
3.1	Contact Center Applications and Vendors	7
3.2	Customer Service within TUI Nordic	10
3.3	Typical Customer Recognition Scenarios	12
3.3.1	Identifying Customer	12
3.3.2	Presenting Customer Information	12
3.3.3	Storing Customer Data	14
3.4	Backend Applications and Flow of Information	15
3.4.1	From Input to Output	15
3.4.2	Constraints of Current Solution	16
3.4.3	Process Overview of Current State	17
3.5	Alternative Approaches	18
3.5.1	CRM Solutions	19
3.5.2	Custom Integration	20
3.6	Omni vs Multichannel	21
4	Recognition Process Design	22
4.1	Hit Rate	22
4.2	Accuracy	24
4.3	Average Talk Time	24
5	Solution Design and Implementation	26
5.1	Technical Environment	27

5.2	Developed Scripts	29
5.3	SQL Code	30
5.4	Improving Accuracy	33
5.5	URL Popup	34
5.6	User Interface Functionality	35
5.7	Color Space	37
5.8	Agile Development Process	39
5.9	Testing	41
6	Results and Analysis	43
6.1	Layout and Constraints	43
6.2	Continuity in Workflow	44
6.3	Hit Rate Improvements	44
6.4	Impact on Average Handling Time	45
7	Conclusions	48
	References	50
	Appendices	
	Appendix 1. Statistics Overview	

Abbreviations

ACD	Automatic Call Distribution
ANI	Automatic Number Identification
CLI	Client Line Identifier
CRM	Customer Relationship Management
CRS	Central Reservation System
CTI	Computer Telephony Integration
DTMF	Dual Tone – Multi Frequency
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IM	Instant Messaging
IVR	Interactive Voice Response
KPI	Key Performance Indicator
POC	Proof of Concept
PSTN	Public Switched Telephone Network
SMS	Short Message Service
SoMe	Social Media
SQL	Structured Query Language
SSO	Single Sign-On
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience
VPN	Virtual Private Network
WWW	World Wide Web
XML	Extensible Markup Language

1 Introduction

TUI Nordic is part of the world's number one integrated tourism business TUI Group. TUI Nordic is the biggest tour operator in the Nordic market with operations in Sweden, Finland, Norway and Denmark. The TUI Group has over 30 million customers travelling from 31 countries to 180 regions around the world. Figure 1 shows the key business figures for the TUI Group, examples of assets like airplanes and cruise ships and figures for the customer base. The group headquarters reside in Hannover, Germany. The workforce is widely spread across the globe, totaling in almost 200 countries at its best (the exact amount of countries depends on the time of year as the business is very dependent on the weather circumstances for its holiday offerings).

The world's number one integrated tourism business at a glance*



Figure 1. TUI Group at a glance [1]

TUI Nordic provides the technical platform for its various brands in the Nordic market. A part of the common platform provided is also contact center applications. The company has recently upgraded their contact center platform to new technology, this has resulted in the currently used application for recognizing the customer not fulfilling all the

new demands. Customer recognition in the contact center environment has been neglected, mainly because there has not been a clear focus of what is needed. The improvement work done has also mainly been designed based on specific input from a minority of the source markets, which has resulted in only small parts of the provided information being actually relevant.

“Taking a customer-centric approach to developing or amending CRM processes will lead to more satisfied customers and generate healthier business results through increased loyalty, retention, and customer lifetime value. Meanwhile, efficiency gains achieved through streamlined CRM processes can help increase customers’ satisfaction and help reduce support costs.” [2,6]

How the recognition process fits into the customer journey and the impact on routing rules is shown in Figure 2.



Figure 2. Routing rules definition flow [2]

The recognition process supports the two first segments shown in Figure 2, these involve identifying and segmenting the contact. The successful recognition in the beginning sets the base for all future routing logic, thus making it very important.

There was a clear need to re-analyze the current customer recognition process in the contact center and based on the outcome develop a modern tool that can easily adapt to future changes in customer behavior. The objective was to develop a process that fulfils the requirements for successful customer recognition within the frame of current contact center application portfolio. The process was then further developed by building a working application plugin, this was then tested within the production environment within the contact center as a proof of concept (POC). The contact channels in scope for the proof of concept were voice, e-mail and web chat contacts. The identification processes was tested only from a customer service perspective, i.e. identification within the web domain was out of scope for this thesis.

According to Peppers & Group [3,7] customers are more likely to continue doing business with a company that exceeds their needs, thus it is also more likely that they will recommend those services to others. Publications from Avaya Inc. [4,14] show that consumers who get good customer service are 70% more willing to spend money with the company. According to Avaya Inc. [4,15] in addition to receiving excellent service customers also want to feel as if they are known.

“The essence of managing customer relationships is treating different customers differently; therefore, the first requirement for any enterprise to engage in this type of competition is simply to “know” one customer from another.” [5,104]

This study focuses on the quick wins that can be gained by creating integrations between different application suites with focus on resolving specific tasks. The goal was to create a working POC setup that was designed based on the outcome of the analyzed data, and test this new design within the customer service functions for the case company. The first part of the thesis concentrates on background knowledge about the principles for recognition and CRM, describing the infrastructure components and software dependencies. Alternative approaches are also briefly described, but the focus remains on the tools available as part of the current application suites within the reference company. The second part describes the developed POC setup, constraints on the design and decisions regarding the underlying infrastructure and developed code. The last part discusses the outcome of the POC, analyzing metrics and giving suggestions on what improvements could be done in the future, both for the basic setups developed within the scope of the thesis, but also giving some more general indications of where improvements could be achieved.

2 Methods and Materials

This chapter describes the scope of the project, how the work was done and which methods were used.

2.1 Context of Project

The task was to switch from the current customer service interface to a new application within the customer service functions at TUI Nordic. At the same time as the contact center application platform was updated also the customer recognition solution was updated to reflect the current needs. Before the implementation started the current state was analyzed, this was done by discussions with key stakeholders and by analyzing existing user feedback. Data from a pilot in Denmark focusing on 24/7 customer service that was done during November 2013 was also analyzed in order to gather background information and to set a baseline for current customer recognition levels.

As a part of the project there was also a migration to new application platform on the backend, this was however not analyzed as these guidelines are set by TUI Nordic IT infrastructure team. However some flexibility in the backend platform still remained, thus the final technical implementation was not yet determined upon startup of project. Customer data retrieval and metrics performed directly on that data were out of scope for this thesis.

2.2 Research Design and Research Process

Initially existing data was analyzed, primarily these were questionnaires done in the customer service departments. Also some interviews were done with the management to get an indication in which direction customer behavior is estimated to go, thus giving a clearer focus on which parts of the recognition process will be key in the coming years. When the basic concept was clear the next step was to investigate what kind of metrics could be adapted to the customer recognition processes.

Research was carried out within the customer service and customer relations management (CRM) fields, thus trying to get a broader aspect of how customer recognition could be done in a most efficient way considering today's rapidly changing customer behavior.

2.3 Planned Research Strategy

The follow up was done in two parts, the first had a limited number of respondents to get an indication if the design and initial measuring works. Then a wider test period started, thus gathering data from a larger number of real customer cases. The pilot period highlighted in which areas there were needs for further improvement.

The final results was also compared to the surveys done before the project started in order to get approval from the management to implement the solution into production across all Nordic countries.

2.4 Overview of Planned Data Collection and Data Analysis

Data from customer service departments in all four Nordic markets was used for the pre analysis. Surveys used include a lot of various data regarding the support/booking process, only some of the data in the survey done before the project start was viable for this thesis. The data that was used mainly focuses on customer recognition hit rates, i.e. how often the recognition was successful and how accurate the received data was. Also average contact handling time comparisons was done before and after the implementation of the new plugin, as the process of feeding customer details into the sales systems takes a decent amount of time for each new contact.

As developing metrics was a part of the project, potential final survey descriptions and measurements were hard to specify in advance. Examples of previously used surveys were also discussed where relevant.

2.5 Development Methods

Development was done according to the agile methodology. A reference group was set up consisting of key stakeholders within the business. Most administrative routines regarding approval processes and input regarding delivered products was managed

within the existing development model for the contact center development team. The reference group within this project did consist of business owners, contact center analysts and where suitable sales agents for initial testing and feedback. There were also external developers focusing on the data mining from the CRM suite, however this was not discussed in detail as it was out of scope for this thesis.

3 Background

This thesis describes how customer recognition is typically done. Also the tools commonly used within contact center solutions to provide user interfaces (UI) with integrations to CRM systems are described. Different alternative approaches are discussed and the key factors to the chosen path are described.

“Customers interact with companies using a wide variety of channels, such as voice, online, Interactive Voice Response, chat, email, or social networking. As their usage of multiple channels increases, so does their expectation that companies should be able to follow their history of interactions across these varied channels, as well as support their needs and preferences regardless of the channel.” [3,5]

3.1 Contact Center Applications and Vendors

Figure 3 shows the different contact channels supported by Avaya Aura Contact Center. Most contact center applications have support for the same channels as these are regarded as industry standards. The main difference between vendors is which features they focus on and try to keep as best in class features. Examples of the different vendor applications and feature sets are shown in Figure 4.



Figure 3. Multichannel contact center [4,157;6;7]

In Figure 3 the contact between clients on the left and agents on the right goes through the contact center application suite. This central part is a compilation of different applications and data stores used to add value to the interaction. This part also makes sure that the customer gets the best service according to company set standards, i.e. pre-programmed flows decide what will happen with the contact. The channels shown in Figure 3 are the most commonly used today [4;30]: voice, internet, video, SMS (short message service), e-mail, IM (instant messaging), fax, social media and the last named "OpenQueue" in the Avaya suite refers to an open API (application programming interface that 3rd party applications or vendors can use to integrate their own special contact types. As an example of the OpenQueue interface could be a ticketing queue system in a retail shop, i.e. the customer takes a physical paper ticket from a machine and waits for their turn to go to a counter to get serviced by a sales staff. Within the web/internet channel there can also be several different contact types: web chat, web callback or co-browsing. These channels are quite common today and usually have their dedicated API's within the contact center application suite, thus making integrations easier. As the OpenQueue works independently from what type of information is transferred via the protocol it also limits the possibilities to add custom features, i.e. everything needs to conform to set standards and any optimizations are done separately for each integration. An ever increasing contact channel is instant messaging as applications like

WhatsApp and Skype bring their IM platforms to the consumers. Usually contact center application vendors only integrate the most common messaging platforms into the IM channel, and the rest will then need to be integrated via the OpenQueue API (or similar if another vendor is chosen).

When comparing basic features shown in Figure 4 it is quite obvious that the application suites are very similar, but have developed custom adaptations for the features they consider most important. One thing to remember when comparing a list of features is also the terminology used, as this might vary slightly between vendors.

	Inbound								Outbound					Social Media			Management		Reporting				Other						
	Call queuing	IVR	Speech recognition	ACD/ANI/CTI	DTMF	Text-to-speech conversion	Skill based routing	Auto dialer	Call list imports	Campaign management	Outbound IVR	Multi channel communications	Multiple initiative management	Social alerts	Customer interaction	Trend monitoring	Integration with social sites	Forecasting & Scheduling	Call scripting	Virtualization	Coaching tools	Call logging	Monitoring	Call recording	Scoring	Database administration	Real time reporting	Automated reporting	CRM integrations
Avaya	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Genesys	X	X	X	X	X			X	X	X	X	X		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Interactive Intelligence	X	X		X			X	X	X	X	X	X		X	X	X		X	X	X	X	X	X		X	X	X	X	X
Mitel	X	X	X	X	X			X	X		X	X		X	X	X		X	X		X	X	X	X	X	X	X	X	X

Figure 4. Comparison of contact center application suite features [8]

In Figure 4 call queuing and skill based routing are probably quite similar features, although only half of the compared products are marked as having the skill based routing feature. When choosing the application suite that is most suitable for the company needs it is usually the design behind the terminology that will have more effects on the outcome than the terminology, thus it can be risky to only compare the feature set lists. The vendors in Figure 4 were chosen from the Gartner Magic Quadrant for Contact Centers shown in Figure 5.



Figure 5. Gartner Magic Quadrant [9]

The report from Gartner shown in Figure 5 is considered one of the more reliable independent categorizations of contact center application vendors. Mitel was included into the example listing in Figure 4 because their presence especially in European markets has grown during the past year. Cisco was left out of the comparison due to their wide range of product outside the contact center portfolio, thus making them more of relevance if the overall IT infrastructure is highly based on the same vendors' products.

3.2 Customer Service within TUI Nordic

The customer service functions within TUI Nordic manage both synchronous and asynchronous contact channels. The difference between synchronous and asynchronous channels is shown in Figure 6. Most of the interactions are inbound contacts, i.e. the

customer has chosen to contact the company, but there is also outbound campaigns used in order to create upsell opportunities or in case of bigger changes that need to be communicated directly with the customer. The contacts can be categorized into two main types: sales and service. The sales contacts are either new bookings or additional services added to an existing purchase. The service contacts are usually questions regarding existing booking or future purchase.

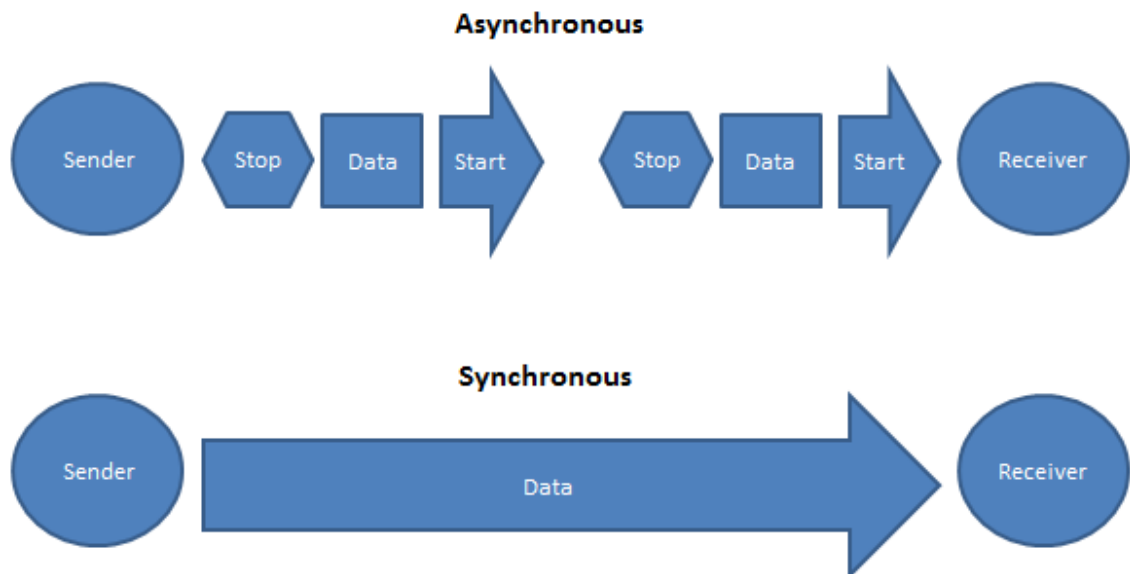


Figure 6. Asynchronous vs Synchronous communication [10]

The contact types shown in Figure 6 are handled by the same sales agents, but with different priority depending on if the communication is done in real time or not. Synchronous communication is happening in real time, therefore the constant arrow shown in Figure 6. Examples of synchronous channels are voice, instant messaging, video conferencing and chat. Asynchronous communication is the opposite of synchronous, this is communication that is not happening in real time. The sender and receiver are communicating in their own pace. Examples of asynchronous communication channels are email, text messaging (SMS), blogs and Facebook [11].

3.3 Typical Customer Recognition Scenarios

This chapter discusses the key factors for successful customer recognition. The chapter is divided into three categories defining what happens prior, during and after a customer contact is handled.

3.3.1 Identifying Customer

The dialogue between a customer and the sales agent starts by a common introductory process, i.e. agent answers the inbound interaction and then the customer presents themselves and why they are contacting the company. Some parts of the information about the incoming contact can be retrieved from data delivered within the CTI (computer telephony integration) connection, thus making suggestions who the customer could be before the sales agent has been able to confirm this. For successfully identifying a customer there needs to be some unique information available that can be linked to a customer id. Usually it is not possible to get 100% accurate results without making the identification process too cumbersome for the customer. Sometimes the data that is delivered before the human contact is not of any use for the identification process, when this happens the sales agent needs to perform the identification process manually during the conversation. According to Avaya Inc. [4,31] 50% of customers think the identification takes too long and 69% say they have had to repeat account details during the same call.

TUI Nordic has chosen to identify the customer based on either the telephone number (voice contacts) or e-mail address (e-mail and web chat contacts) that is linked to a customer specific id. This data can be observed without any additional efforts from the customer perspective, thus making it invisible for the user. This approach has been deemed good enough by the business as the customer cannot get access to any sensitive information via these contact channels, thus there is no need for strong key-pair authentication.

3.3.2 Presenting Customer Information

When a customer has been identified the next step is to present data that has been linked to the current customer card to the person who is assigned to handle the con-

tact. TUI Nordic uses Microsoft SQL Server database engines to store and provide customer data. As described in the previous section a customer id is processed and based on this key data is retrieved from the CRM data source. The data presented to the sales agent is decided based upon company metrics and the relevance of the data in order to complete the customer interaction efficiently. In Figure 7 the flow of data is shown from the point where the contact arrives until the data fetch returns the desired information into the client application.

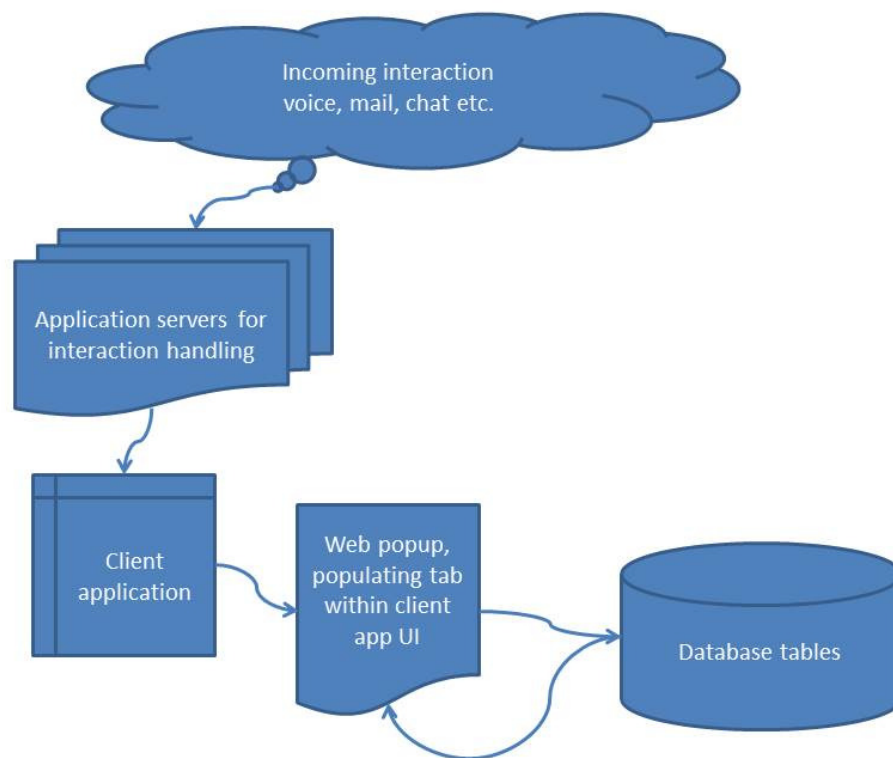


Figure 7. Interaction popup flow

The application servers shown in Figure 7 are e.g. communication gateways for voice calls, e-mail servers and other web or related application servers. The client application launches a popup within the user interface where the html webpage plugin is loaded. This html page then executes scripts that make lookups into the database tables where the underlying data is stored. The returned information is then presented within the originating client application in the dedicated tab.

3.3.3 Storing Customer Data

In addition to only showing the customer card as described in previous sections there is also the possibility to correct any data that is wrong. As legislations vary in each country on what type of data can be stored and for how long, there needs to be restrictions on which fields of the customer card are editable. This thesis does not describe the limitations put on data storage from legislative approach as it would be very time consuming as usually all countries have their own specific regulations. The legislative limitations are also not important regarding the POC this thesis work relies on. Figure 8 shows how the “save updated data”-routine works.

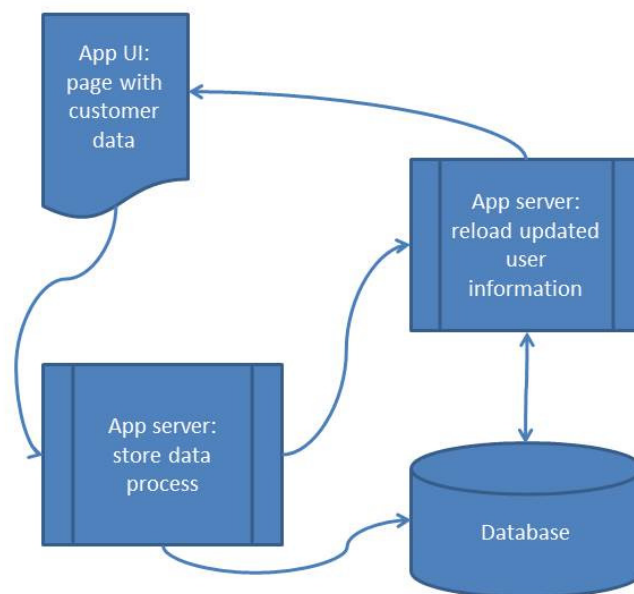


Figure 8. Store data process flow

As shown in Figure 8 the data is firstly stored, and then the user interface is reloaded with the updated remarks corrected. This flow is designed with the intention for always having the latest info available, and also to minimize risk of incorrect duplicates as the data and views are constantly kept up to date when changes are occurring.

While storing data it is crucial that duplicates are to be avoided, thus there needs to be certain logical steps taken while processing the update requests. In the SQL (structured query language) server data store there are built in functions for basic locking

features. As the update process of customer data only consists of one active connection at a time (it is safe to assume that e.g. one customer cannot call several simultaneous calls at the same time from the same phone number), the built in features are enough to keep data from becoming corrupt due to simultaneous updates.

3.4 Backend Applications and Flow of Information

This chapter describes how the data is refined when going through different backend systems. Current technical constraints are discussed and what limitations there are on the currently available data.

3.4.1 From Input to Output

As customer data can be added either by a sales agent or by the customer themselves on a website, it also means that the accuracy might vary. There is always the risk of inserting typos into the data, which will then affect the hit rates when the data is processed by the lookup queries. To avoid common mistakes it is recommended to analyze the input fields in order to at least get the right kind of information stored, i.e. phone numbers should not contain letters and e-mail addresses need to comply with a specific format. This could be done by automatically checking the input fields in web forms and implementing similar functionality to other applications. When the data is to be used in several backend systems it is also vital that the format is documented and that all input fields comply with the chosen data types. It is also good practice to do any simplification of data in the presentation layer, e.g. shortening the presentation of a datetime value to only show the date part.

TUI Nordic gathers all customer data into a custom built CRM solution, either in real-time or by imports via nightly batch jobs. Every night there is also a clean-up of the data and business analytics applied to map data in the desired way. As there is a lot of data that is not of relevance for the customer recognition process it was decided to extract the relevant information from the core CRM data store into its own dedicated database with specific tables for the recognition process. In this way the access to sensitive data in the CRM can be limited, as the recognition part only needs specific parts of the stored customer information. This also circumvents some of the limitations put by

local legislation regarding handling and storing data which includes personal information.

Customer information can be accessed via the CRM tool, but as this is a multifunctional application the search process can be tedious, especially if it needs to be done manually during a phone call. In order to provide the sales agents with identification data already at the start of a voice or multimedia conversation there has been implemented a website that does lookups into the database based on the input triggers sent from the contact center application. This approach gives a good combination of speed and accuracy, and when the database is frequently updated with correct data also the hit rates keep getting better. The layout also mimics other applications in use within the company, thus making it easy to get used to. As the layout uses documented design standards for UX (user experience) it is also easily modified in the future if visual redesigning is desired.

3.4.2 Constraints of Current Solution

What are the current limitations and how is the business process changing? The old solution in place had several limitations that should be overcome, and also some of the data presented for sales staff was not relevant anymore as the data was related to old systems. Some of the issues were easier to change than others either by removing data from the views or improving performance and load times on the server side, some were not feasible to modify at all. One of the biggest questions was if there is a possibility to integrate directly with the CRM tool, as this would have given all the functionality without need for dedicated middleware. Unfortunately there were already performance issues in the current CRM tool and in addition also the access limitations to certain parts would have been cumbersome to overcome. Another big issue was hit rates and data accuracy, as it often seemed that the customer info was not always showing the latest updates. During the due diligence phase it was also found that some customers were logged with several customer cards, but with inconsistent information and thus making it impossible to filter and clean-up by automation.

There were also performance and reliability issues in the lookup phase. It was therefore decided that also the hardware platform needed an overhaul. Luckily the core of the contact center solution was recently upgraded to a virtual environment, making it more easily modified and adaptable to the new web based plugin design that was de-

veloped. The rest of the old recognition setup was relying on physical server hardware that was close to end of life, and as virtualization tools were available also this part was upgraded to current standards.

3.4.3 Process Overview of Current State

The initial workflow consists of several tools that are used by sales staff in order to get accurate information and to be able to register bookings. As the tools used for interacting with the customer are different from those used to search and book trips it easily leads to staff only using the vital parts and not all available tools for customer management. One of the tedious tasks in the process of getting customer information is the manual lookup phase within the CRM tool. As the search can be done in various ways it might not always be easy to find the best way that will result in the most updated customer data. Also the limitation of showing only one customer card at a time makes the identification process slow, which often leads to skipping this phase and just asking the customer for their personal details every time they are in contact with the customer service department. Both manual search and asking for basic details every time result in longer talk times and thus less handled contacts per hour. Figure 9 shows the initial workflow for a contact center agent.

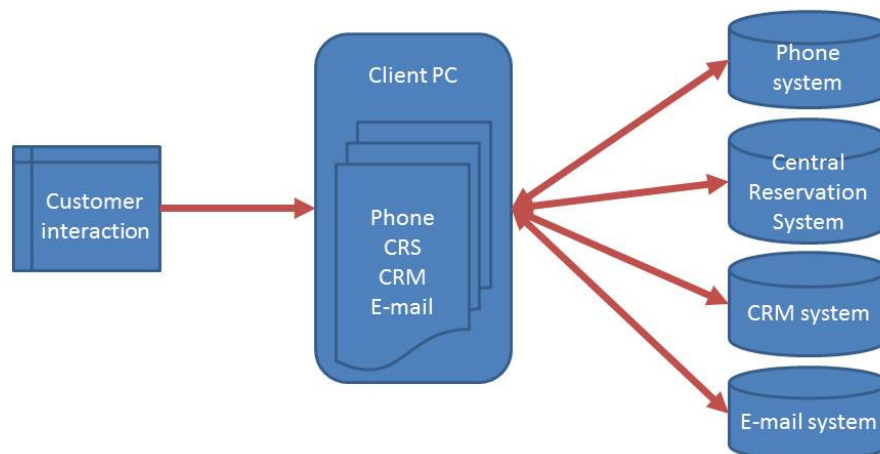


Figure 9. Old process flow

As can be seen in Figure 9, there are several applications running on the client PC in order to get all relevant information regarding the customer interaction and to then be able to process the service request. The more programs that are running simultaneously the more computing power is needed from the PC and the risk for opening and closing incorrect applications increases when the icons in the taskbar are getting smaller. Also each manual lookup phase takes some time to complete and deliver data to the client application. The possible savings in upgrades of the PC hardware used by agents was not evaluated in this thesis as it would have had too many influencing parameters, however it was considered an added bonus if the amount of simultaneous applications needed can be decreased. There will still be some specific tasks and workflows that require the full toolkit of applications to be used also in the future, but as those are not the most time consuming ones those were neither analyzed nor addressed in this thesis that aims for a simple POC setup.

3.5 Alternative Approaches

The Avaya Aura contact center platform used in TUI Nordic has good possibilities for custom integration. The two main alternatives for integration were a) using a lightweight version of current CRM tool, or b) developing a web-based plugin that integrates to the contact center application and loads data from an external data source. A third option would have been to analyze the whole CRM portfolio to see if it meets current needs, but as this platform is linked to very many applications it was decided the workload for this is too high in comparison to the issues it was intended to solve. It was however noted that for future improvements a detailed analysis of the core CRM solution should be initiated as an own project, in this way also other stakeholders can be involved and the outcome would hopefully be a more integrated system which then would minimize the need for 3rd party applications and integrations in the future.

Figure 10 shows an example of the out of the box feature for integrating a web URL (uniform resource locator) into the Avaya Aura Contact Center desktop application.

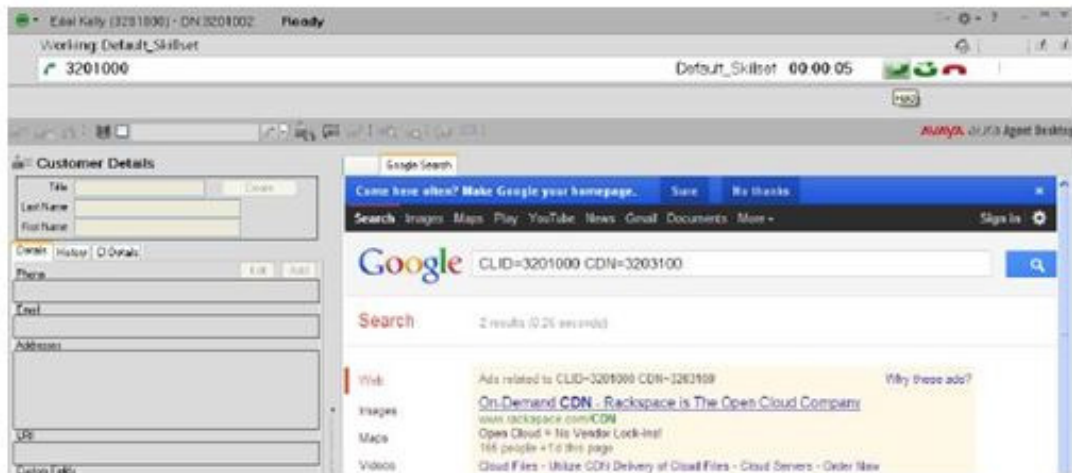


Figure 10. Example of webpage integrated into contact center application [6]

Figure 10 shows a Google search page integrated within the contact center desktop client. In this example the parameters used for the URL lookup are basic information found within the incoming voice call, but this can easily be adopted into the correct business requirements for different contact channels.

3.5.1 CRM Solutions

The obvious starting point for integrating customer data into the recognition process would be to look at existing CRM solutions and out of the box functionality within these. In the chosen customer case there is unfortunately not any of the usual CRM application suites in use, some of the well-known suppliers include Microsoft with Dynamics, Salesforce.com and SAP with various products. As many companies nowadays have very scattered data stores and backend solutions there is not always one central point of data store to rely on. The current CRM data store used within TUI Nordic is a customized application developed for TUI Nordic by a 3rd party. As this application was originally developed primarily for other purposes than to be used as a CRM tool for customer service departments there are many constraints that make it complex to use in customer facing functions. There are also legislation issues that prevent the usage of current client application in all parts of the customer facing interactions. In general many CRM solutions used within global companies today have many local customizations, thus making it difficult for contact center vendors to support all CRM suites. The big out of the box solutions like Microsoft Dynamics and Salesforce.com are usually within scope for contact center application vendors, but the other players in the market

are usually left with the standard open API's to customize their own integration. This often leads to some diffusion of what can and cannot be accomplished as the stakeholders regarding CRM and contact center usually have different focus. Figure 11 shows examples of the out of the box integrations available.

Supported CRM integrations					
	Salesforce	Sugar CRM	InfusionSoft	Microsoft Dynamics CRM	Netsuite
<i>Avaya</i>	X	X	X	X	X
<i>Genesys</i>	X	X	X	X	X
<i>Interactive Intelligence</i>	X	X		X	
<i>Mitel</i>	X	X		X	

Figure 11. Example of standard CRM integrations [8]

As can clearly be seen in Figure 11 there are commonly supported application suites that most vendors will support, and then some additional suppliers that are probably working closer together with integrations only with some of all the application vendors.

3.5.2 Custom Integration

As noted in previous section the current CRM tool cannot be integrated directly to the contact center platform, thus it was decided to evaluate the possibility to create a middle layer between these application suites in order to get the most crucial functionality without the need to invest in completely new applications. The integration steps will be discussed in more detail in the solution design chapter, they consist of a) data export from CRM and b) plugin for Avaya Aura contact center client application.

The benefit of a custom integration with the help of a middle layer is that all components only need to interact with one other part, thus development can continue in each application independently of each other. If any software or applications change everything will continue to work as long as the middle layer is updated. Changes to frontend and backend systems can be done dynamically, thus enabling developers to focus on the parts of the solution that integrates to their application suites. The risk with having split responsibilities is that a change or problem at any stage might not accurately be noticed at a later stage, thus leading to longer fault investigation and correction proce-

dures. This risk was deemed minor, and the effect on the final product (the webpage plugin) was also considered to be of non-importance.

3.6 Omni vs Multichannel

There has for a long time been a trend in contact center environments to process contacts from various channels, this has been called multichannel. Recently another acronym has been widely adopted as the hype term to use, this is called omnichannel. Both definitions are based on an environment where various contact channels are bundled into the same solution, where the main difference is that omnichannel takes this a step further from basic multichannel. In multichannel environments interactions are looked at as different contact types, and thus given different treatment. Within an omnichannel environment all interactions are looked at as contacts, and the routing is done independently of contact type. As the wording is quite close to each other it can sometimes be confusing to separate them, and thus also many industry influencers are not making a difference between these two wordings at this time.

“Omni comes from the word Omnis which can mean all or universal. This is in comparison to other categories out there, like “multichannel”, from the Latin word Multus, meaning multiple or many and from crosschannel, derived from the Latin word Crux, meaning to go across. The way that many are explaining omnichannel today is: ‘cross channel being done well’.” [12]

When the contact channels increase this can also be seen as a positive thing for the contact center agent, when the agents’ daily job varies between multiple channels it might help to avoid fatigue and increase employee satisfaction [4,122]. Multiple channels can also give the customer a better way of adapting their chosen channel between the real-time and non-real-time ways to get things solved [4,137].

4 Recognition Process Design

What affects successful customer recognition, how can the odds be improved and what are the risks of getting too many positive hits? The main factors to think of when designing a solution to use for recognition purposes is to have a clear definition of what a customer is and are prospects treated equally as customers that have already invested money in the company's products. If a company uses a customer award program this identification step is quite easy to define, but if all contacts should be treated equally it gets harder to accomplish accurate recognition. The downside of getting many hits on a search is the risk of returning incorrect values, thus losing some of the finesse in having the information to recognize the customer and make them feel important. When a solution results in a good hit rate with enough accuracy the impact on the average handling times for contacts is the most obvious positive result.

4.1 Hit Rate

The easiest way to monitor if recognition is performing well is to measure the hit rate. A hit rate is the number of successful scenarios for each time a contact arrives, the more results that are returning data the better the hit rate is. The hit rate can also be improved by asking the customer questions prior to delivering the contact to a sales agent. On a website this can be easily done by using mandatory fields, but there are still limitations in the verification logic that can be used to verify the user input. A similar approach can also be used on incoming phone calls, by using IVR (interactive voice response) systems to ask the customer for identification information in the beginning of the call before it is queued to the sales agent. Too complex IVR's are usually not recommended as the risk of customer retention grows when they feel like the majority of the time is spent talking to a machine. Adding recognition to the IVR setup might also clutter the flow of calls, as the time spent before actually having the possibility to get connected to a real person increases.

A good approach to increasing the hit rate without affecting the length of phone calls is to use the CLI (Client Line Identification) information transported in the voice stream from the PSTN (Public Switched Telephone Network). As customers use mobile phones more than fixed line phone numbers this is resulting in quite good accuracy if the phone number is used as identification id. For multimedia contacts like e-mail and

web chat almost the same accuracy can be achieved by having a mandatory field to input the customers personal e-mail address, or reading this information from the mail server that relays the incoming e-mail contacts to the contact center environment.

There are always contacts that can't be connected to existing customer information, although they would already have been in contact with the company earlier. Some customers might have changed or previously logged incorrect contact details, calling from hidden phone numbers or via a company extension. The ratio between new customers and returning customers also highly affects the hit rate, as the potential of identifying the customer is higher in businesses where the amount of returning customers is better. E.g. if half of the business comes from new customers then a 100% successful recognition would give a hit rate of 50%, but if the ratio then goes more towards customers that are returning also the possibility to increase the hit rate grows.

The baseline hit rate was analyzed by doing a survey among sales staff about how often was the customer recognized in the old solution. Figure 12 shows the split between Yes and No answers. The total amount of responses was 6388 contacts, where most of these were phone calls. As the setting for the survey was a simple Yes or No question there was also no risk of misinterpretations how to log the contact.

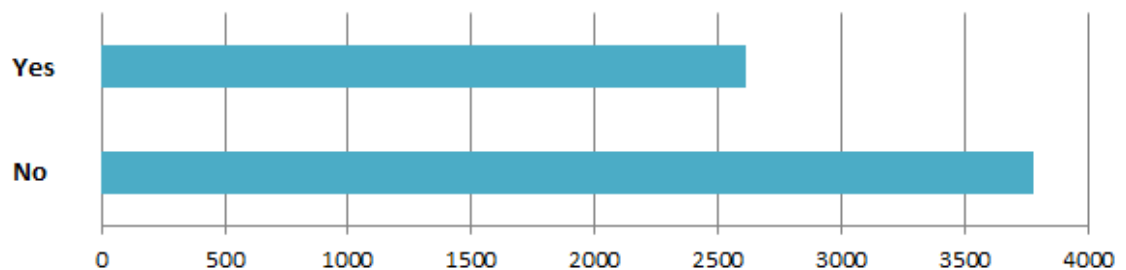


Figure 12. Hit rate in old solution

Based on the answers in the pre-study only ca 40% (2613) of contacts were recognized by the old system. When this information was discussed with the reference group it was clear that the amount of successful hits should have been higher, this was based on the view within the group that the amount of returning customers should be higher. The target was then set to improve the hit rate from a 40-60 ratio to the inverse 60-40 ratio. The hit rate will also vary to some extent depending on the amount of returning customers, however analyzing customer retention is out of scope for this thesis as it is mainly affected by other means and measures. The potential to good customer service

with the help of recognition is however greater when the amount of returning customers is higher, this is mainly due to the fact that the more data can be collected and linked to a single customer the better the recognition process will work and thus give the customer the feeling of being recognized and valued.

4.2 Accuracy

As described in the previous section a successful hit rate is the primary key to succeeding in recognition, but one thing that often is forgotten is that the provided information should also be as accurate as possible.

As the most important thing is to have relevant data it was decided to try to clean up the initial database in order to remove entries that did not seem legit. The problem with having both test data and other dummy entries in the source database was a known issue from previous analysis within the CRM team, and as part of the new recognition process it was deemed fit to finally do something about that issue. The faulty and partially duplicate entries had been populated into the source databases for a long time as system migrations and integrations had left data used within internal system testing on the servers. Also some of the incorrect information could have come from web based systems that had lacked data verification procedures before triggering updates into the central data store. The cleanup process will however not be described in detail in this paper as the source data comes from a CRM system that is out of scope for this thesis.

4.3 Average Talk Time

As discussed in the previous section accuracy is the key for succeeding in recognition. If the system performs fast and accurately this should then lead to shorter overall handling times for the customer interaction. The new design was compared with statistics gathered before the implementation from a limited set of interaction flows, thus aiming to demonstrate the effects of the process and redesign efforts. As average handling time is widely used within the contact center framework there are of course also other things that affect these figures, thus setting the time periods to be comparable might be difficult. In this analysis the figures were compared for a period of half year, where the data was gathered 3 months prior and 3 months after the launch of the redesigned

recognition plugin. By taking into account a longer timeframe any seasonal anomalies should have less effect on the outcome. The overall trend for talk times is fluctuating throughout the year depending on the incoming call volumes, but also marketing and sales activities impact on the type of contacts being handled and thus also talk times.

5 Solution Design and Implementation

As the gathering of data varies depending on what source is used for input this chapter only describes the components specifically used and designed for the proof of concept setup. Figure 13 describes the data lifecycle, but where the input stream has been simplified to only one unified model. This was done in order to not spend time on describing wide and complex data flows that were not of interest for the actual recognition process.

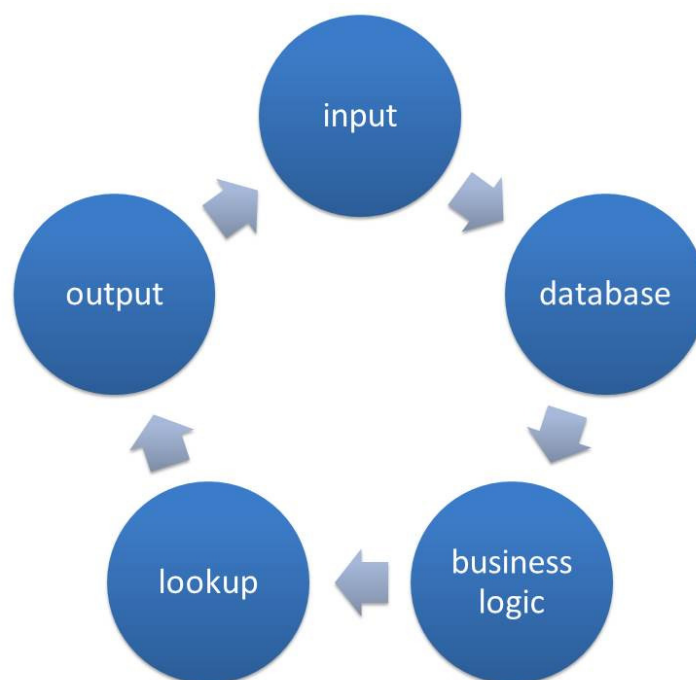


Figure 13. Data flow

In Figure 13 the input phase is associated to entering customer data via any of the available contact channels (internet, customer service etc.). The data in the first phase might look different depending on from which system it originates, however some of the key components for customer recognition need to be present in order to link data to a specific customer card. The next phase is to store the data into a database, this might be one or several systems depending on what type of information is stored and where it is intended to be used in the future. The third phase in Figure 13 is one of the more important ones when it comes to successfully recognizing the customer. In phase three specified business logic is applied to the information stored. When the data has been tagged in the correct way the lookup phase can then provide the user interface with the

correct output information. The last phase from output to input then concludes the cycle, where the original data can be updated with more information or incorrect data modified to include the right information, and thus providing better relevance in future search process. The most important steps for getting relevant data into the database are the input and business logic steps, as these have access to all data present and can make alterations based on the overall picture. One example of what the business logic step could do is link the customer data against company metrics in order to calculate a value for each customer card. Also specific repeating customer behavior could be analyzed and linked together in order to achieve better customer interactions during the next contact.

5.1 Technical Environment

The technical setup was designed upon virtual servers within the existing hardware platform in order to keep running costs down. Utilizing the same environment to run several different tasks is also a more eco-friendly way of running a datacenter, which is very important for an organization that has set high sustainability targets. One of the benefits when integrating solutions by internal resources is wider options to adapt to various technical platforms, if the same setup would have been sourced from an independent contractor then those contracts would probably put constraints on the technical environment that would require dedicated hardware resources. Figure 14 shows the hardware design.

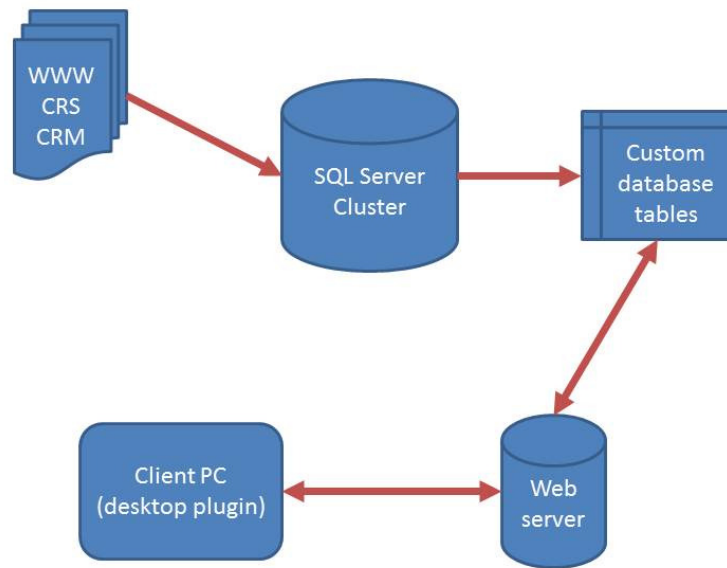


Figure 14. Hardware design

The main components in Figure 14 are the database server and the HTML (hypertext markup language) webpage residing on the web server, the rest of the solution could be replaced by any similar configuration. The client side specifications are quite simple; built-in web browser or similar client that supports HTML and CSS (cascading style sheets). The clients could have been configured to access the data directly, but as it is more secure to use server side scripting for accessing information from SQL server databases this was decided to be the better option. Also load times on the server side are faster than if queries would be run from the client over the data network. It was also desired to limit connections from too many sources to the database environment, thus accessing everything via the server proxy it makes data security and logging easier.

During the development phase it was not stressed that the HTML code used in the POC should work device independently, thus no special adaptations regarding screen size was made. However the HTML code used during testing should work on a wide range of devices reaching from PC/laptops and tablets to smartphones. The restrictions in the network infrastructure however limit the access to the server side components, thus the webpage was designed to only work on company internal data network. As this is running completely inside the company firewall it also makes data security better. Limiting the code to run on internal network only was also in line with the work pro-

cesses this plugin supports, i.e. staff utilizing the plugin rely on other tools that are only available on the internal data network (either connected directly from any of the offices or remotely via VPN (virtual private network) connection).

5.2 Developed Scripts

The most visible part that was developed was the scripts for the webpage. The core consists of a basic HTML page that functions as a template and then a script file that holds the logic, the flow is visualized in Figure 15. A third file for design purposes was considered, but as the template was quite basic it was chosen to include the CSS formatting within the HTML template file. However if the plugin would have included several HTML pages centralizing the CSS code to a single file would have been a better approach. The chosen format for the files and code is .aspx and C#. As the features used within the code are quite basic it was chosen to continue with the same frameworks as used in other components integrated into the contact center applications and plugins, this was mainly because the simplified management and future integration possibilities.

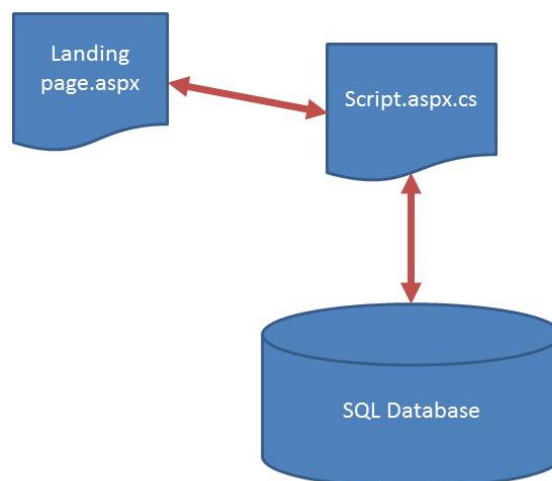


Figure 15. Scripts

The initial HTML page is loaded either as an empty container or then with pre-defined parameters (if customer info is available). The call to the script file is always initiated from within the container template. The script code is always executed with parameters, as no part of the code is useful without having information lookups for a base for recognition. When the page is loaded, either with info from the server or as an empty page, the next thing is to set the layout and design. There are also some features used for aiding the user to find and use the information at hand, e.g. pre-selecting data and highlighting latest updates etc. When the first page load is complete there is also the possibility to manually do new lookups via the Search-function, this is developed to reuse the same code as used within the initiation process and thus makes maintenance and monitoring easier when everything is consistent.

5.3 SQL Code

Most of the finesse regarding the SQL data structures and the business logic implementation is done already in the data transfers from the underlying CRM and data stores, but there is still some SQL code included in the developed plugins. For security purposes it is usually recommended to develop stored procedures and then call these predefined functions from the application. According to A. Kriegel [13,134] stored procedures can improve application performance and also reduce network traffic.

“Another important benefit of stored routines is code reusability. Once designed and compiled, a stored procedure” ... “can be used over and over again by multiple users (or applications), saving time on retyping large SQL statements and reducing the probability of human errors.” [13,135]

The SQL code used by the plugin is quite basic as it was developed only for use in the POC. The basic consists of SELECT queries with JOINS to access data in several tables. The data is stored in three different tables in order to retain execution performance for the queries. The modification of data is allowed only into one of the tables, thus making the underlying core information flow more secure and less risk for outer intervention. However the concern for data security and attack vectors like SQL injection are quite complex, thus they are out of scope for this report as the POC is not meant to withstand intrusion testing.

Some minor performance tweaks were done to the SQL queries after initial testing phase. This was mainly done to achieve better performance when the amount of data to process increases. One of the things added was creating an index in order to get a better data structure and thus faster execution times.

“Index significantly reduces disk I/O and logical reads, to boost up the performance of the SELECT statement by locating proper data without even scanning the whole table. That is why it is mandatory to have a proper index on proper column(s) of the table. Missing indexes or Indexes on improper column(s) could start creating performance-related issues, such as implanting a wrong execution plan, which may create high I/O use and logical reads.” [14,197]

The basic structure of an index is shown in Figure 16.

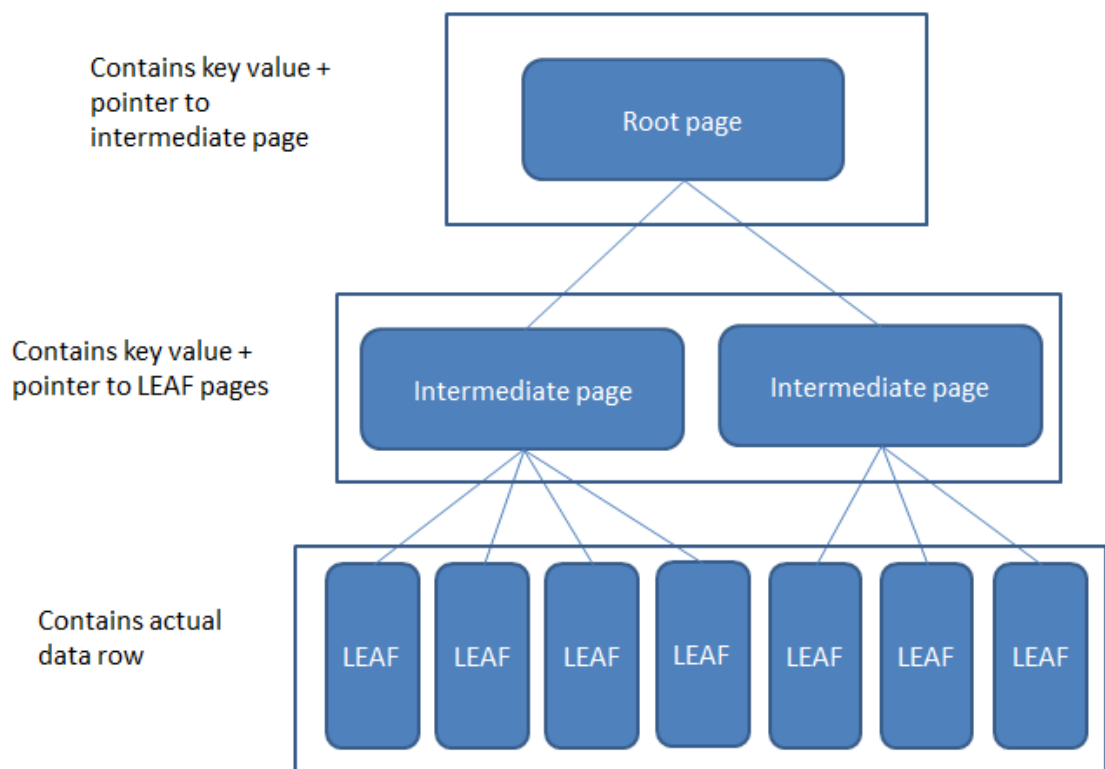


Figure 16. Basic structure of an index [14,198]

The intermediate pages shown in Figure 16 contain the pointers to the actual data, thus making searching and accessing data faster.

Listing 1 shows the index code. It was chosen to use the most viable column going forward, as the multimedia contact types are increasing the index was based on the e-mail address.

```
CREATE INDEX IX_Email ON [Data-
base_Customer_Service].[dbo].[dropoff_customer]
([email])
```

Listing 1. Example code for index

The addition of the index shown in Listing 1 resulted in better performance for read queries. In Listing 2 the new table structure performance with indexing is compared against the old single table structure.

Object Name	Missing Index	IO waits	IO wait time (ms)	Avg IO Wait (ms)	Range Scans	Index Lookups
[dbo].[dropoff_customer]	Yes	6433	27858	4,33	184041	4233973
[dbo].[dropoff_reservation]	No	5311	24619	4,64	1532	4909167
[dbo].[dropoff]	No	139	3009	21,65	117	4496994
[dbo].[dropoff_customerlog]	No	2	15	7,50	813	0

Object Name	Missing Index	IO waits	IO wait time (ms)	Avg IO Wait (ms)	Range Scans	Index Lookups
[dbo].[dropoff_customer]	No	20343	64574	3,17	282056	5821823
[dbo].[dropoff_reservation]	No	21367	101013	4,73	27566	14537313
[dbo].[dropoff]	No	75	299	3,99	179	4498977
[dbo].[dropoff_customerlog]	No	2	0	0,00	1364	0

* only the first 2 tables are relevant as the others don't have enough volatility to give correct numbers

Listing 2. SQL statistics

When comparing the average IO wait times for the old dropoff table against the new tables that were split into dropoff_customer and dropoff_reservation it is quite clear that the wait times are drastically improved. The average has gone down from around 22ms to values around 4-5ms, which is over 20% faster response times than in the old process flow. Also the wait times for the third table dropoff_customerlog are clearly lower than the old single table structure, however this is a new function that was not present in the old process and thus the response times cannot be included in the overall comparison.

Listing 3 highlights the change for two example tables. These examples were chosen to represent both the normal congestion when the amount of data grows bigger and the impact indexing has on the performance.

<u>Item</u>	<u>Avg IO Wait (ms)</u>
cust_old	4,33
cust_new	3,17

Change: 27 %

* shows the performance improvement related to the non indexed table

<u>Item</u>	<u>Avg IO Wait (ms)</u>
res_old	4,64
res_new	4,73

Change: -2 %

* indicates the normal development for the increased data

Listing 3. SQL statistics comparison

In the examples in Listing 3 the average IO wait times show how indexing can make a difference in improving performance. When comparing the average wait times before and after indexing the performance increase is +27% (cust_old vs cust_new). When this is then compared to the other reference column that was already indexed (res_old vs res_new) it shows that the normal decrease in performance when the data grows would have been -2% slower response times.

5.4 Improving Accuracy

In the proof of concept setup it was found that many of the factors available to be used as key values in the lookup process were ambiguous, thus resulting in more than one unique result. In order to get the most accurate results and improve the hit rate the following was done to the data:

- Cleanup of core database
- Creation of business logic to determine which result is most accurate when performing the lookup
- Review the results and fine tune logic

The next step was to analyze why there still were queries that returned more than one customer card as result. After digging into the data the analysis showed that these were actually correct results, but in order for the recognition to be more accurate on the

first search the results needed to be weighed against each other. The principle for the algorithm used for this was to analyze the customer card and compare how the different fields compare against other cards in the result. Usually the card that was last updated should be the most accurate, but as testing proved this was not always the case. However when also combining some of the historical booking data and various contact points to the analysis the process was finalized and deemed well enough after initial testing. The addition of an automatic feedback loop was discussed, but it was decided not to be implemented as the core reason of this issue actually resides outside the scope of the recognition process, thus this was also given as feedback to the CRM team to add to their list of future improvements into the source data.

In order to be able to actively monitor the performance of the developed plugin the following metrics were added to the plugin page:

- Number of total page loads
- Number of successful hits
- Number of manual searches

These parameters are stored within a simple .xml file on the application server in order to easily be able to reuse the data later in other applications if necessary. The first value tracks all incoming contacts independent of channel, then it is categorized as a hit or miss and the correct value updated. The last value tracks if the agent does manual searches, which is mainly included in order to see if the automatic search function is performing as it should or if the manual searches are giving better results.

5.5 URL Popup

The integration of the plugin website into Avaya Aura Contact Center Agent Desktop application was done via a template that adds the website URL as a popup within the standard desktop application container. Multimedia Templates shown in Figure 17 is a standard feature within the Avaya portfolio and was used already within TUI Nordic for adding other websites into the default agent desktop user interface. By utilizing the integrated web browser component of the contact center desktop application it was easier to create continuity in the workflow, and also at the same time removing the need for yet another open web browser window.

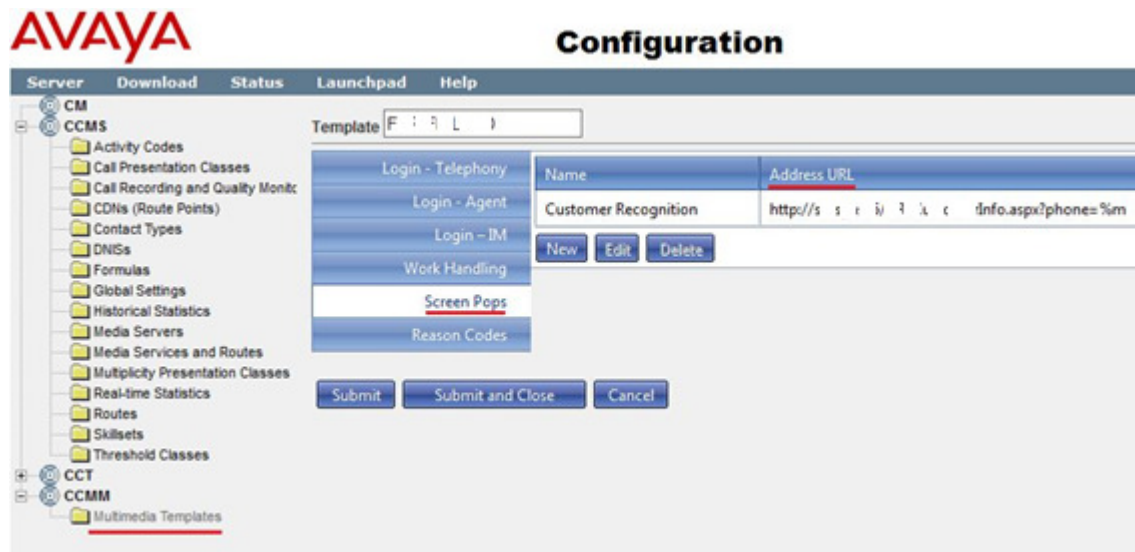


Figure 17. Screen Pops feature

Figure 17 describes the configuration of web popups within the Avaya Aura Contact Center application. As the recognition only requires one parameter the URL is quite simple, but if there would be need for multi-parameter usage that is also possible within this application. As the access to data is based on SSO (single sign-on) principle no added authorization mechanism was added to the client website, but as the agent desktop application requires a windows domain account for authentication before it will start the plugin was deemed secure enough.

5.6 User Interface Functionality

The plugin that was developed as a proof of concept to demonstrate previously discussed ideas and concepts was divided into three main parts plus an additional research possibility. Figure 18 highlights the different parts, the chosen details and fields used in the plugin are there only as examples because the dynamic design allows for easy customization according to evolving business needs.

The screenshot shows a web application interface with a header bar containing a logo, a search field, and a green bar with the number '1000'. The main content area is divided into three sections:

- Section 1 (Left Column):** Customer identification fields including Birth date, Address, Customer ID, C/O, Customer Status, Zip code, City, Home phone number, Cell phone number, Blue Card, Email, Latest claim, and Reservation ID/Date.
- Section 2 (Right Column):** Reservations and History tables.

Reservation	Date	Destination	My Trip visit date
.....	2015-..-..	MBJ 08:38:00

Reservation	Date	Destination	Accommodation	Room type
.....	2014-..-..	MRU	AMB	DR ST
.....	2014-..-..	IBZ	SIN	DR NO
.....	2014-..-..	CUN	AZT	DR NO
.....	2013-..-..	KGS	GOO	DR NO
.....	2013-..-..	BGI	ALC	DR NO
- Section 3 (Bottom):** Conversation log with a text input field, an 'Add' button, and a table of log entries.

Date (CET)	Comment	Agent
2014-05-25 10:13:42	test log entry from production server

Figure 18. Webpage User Interface

The most important part is found both at the top of the page and also in the left column (nr 1). This information is directly related to the recognition process and is the foundation for all other data retrievals. The first part consists of the customer identification information, added with some business metrics in order to help sales agents process the contact. The second part in the right column (nr 2) shows information about customer history. The third part resides at the bottom (nr 3), showing most recent comments added by sales agents. As the due diligence indicated that there is a risk of not getting the right data on the first search, or that at times the sales agent needs to lookup a customer manually, it was chosen to add a Search-field in the top right corner in order to catch customers that were missed in the initial search.

The functionality within the developed web plugin shown in Figure 18 adheres to the company guidelines and design principles for web development. It was also important to keep a similar look and feel in order to have an intuitive user experience. The most important UI features regarding the recognition process was the presentation of customer data and the Search-field. As the process starts off with automatically looking up data based on CLI or e-mail, the basic view is always pre-filled with information. If this is not the correct information then by inputting new search criteria in the search-field will restart the lookup process. The lookup criteria are exclusive, i.e. if there are several

identification entries that match the search it will then proceed to analyze the most recent entries and return data for that customer id.

The conversation log at the bottom of the page is the only one that has a different lookup process regarding to the other parts of the plugin. After analyzing the customer data in the database during the development phase it was decided that the log should not do exclusive lookups on only one criteria, i.e. it will populate information based on any of the identification criteria (CLI, e-mail or customer id). This decision was made as a compromise to accompany business requirements and to be able to group customers in specific cases. As this is not optimal for data integrity, but it was deemed good enough as the existing workflow could then be retained.

5.7 Color Space

The starting point for layout and color schemes was that it should blend in the contact center application without standing out too much. As recognition is a process that is done in the background by the system it was chosen to reside within its own tab in the contact center application, this is shown in Figure 19.

The screenshot displays the Avaya Agent Desktop interface. At the top, there are window titles: "PatrickM501100:023", "AACC: Not Ready", and "Elite: Not Ready". The main window shows a search bar with the text "00:00:26" and the "AVAYA Agent Desktop" logo. Below the search bar is a "Customer" sidebar with fields for "Title", "Last Name", "First Name", "Phone", "Email", "Addresses", "URLs", and "Custom Fields". The main area is titled "LIME Recognition" and contains a search bar. Below the search bar are several tables: "Reservations" with columns "Reservation", "Date", "Destination", and "My Trip visit date"; "History" with columns "Reservation", "Date", "Destination:", "Accommodation", and "Room type"; and "Latest claim" with columns "Reservation ID" and "Date". At the bottom, there is a "Conversation log" section with a text input field and an "Add" button.

Figure 19. Web Plugin within Avaya Aura Contact Center Desktop Agent

Different background colors were tested during the development, from grayscale to company branded coloring. There has been a lot of research regarding which colors work well together, both regarding contrast and familiarity. Many researchers agree that black text on white background is better than white text on black background [15].

Some recommendations for different contexts from literature studies:

- For sites where retention and especially readability, are a major concern the combination of text that should be used is black on white or a closely related combination. Both the contrast ratio (black and white) and the fact that we are familiar with black text on white seems to give an advantage over the opposite white text on black (the contrast is the same, but it is less commonly used) that was rated much lower on readability. Therefore, if other color combinations are the convention for a given context, then the convention should weigh as heavily in the decision as contrast. [15]
- A readable site is usually also viewed as professional, so if “professional” is a desired image to be portrayed these same readability guidelines should then be applied. [15]
- For sites where aesthetic and purchasing behaviors are a major concern, colored text and background combinations should be preferred. The site is seen as more visually pleasing and stimulating when chromatic colors are used. These colors are also more likely to lead a viewer to the intention to purchase products advertised on the site. “Combinations involving the color blue, and including two chromatic color (e.g., light blue on dark blue) appear to be preferable to a combination with less contrast and including a chromatic color (e.g., cyan on black) for promoting positive affect and behavioral intention.” [15]

“Contrast affects legibility, but unfortunately, it does not seem to be as simple as high contrast being better than low contrast. In the main experiment, green on yellow had the fastest reaction times (RT), and in the control experiment, medium gray, and dark gray had the fastest RT's. In neither experiment did the black on white condition show

the fastest RT's. These results show that these participants had faster response times when more median contrasts were used." [16]

The readability of text is not affected only by choice of colors, but also styling (plain, italic or bold) and the chosen font influence the way we interpret things. There is not a specific combination that is always best, but rather some fonts work better in certain color combinations. [16]

In the end it was decided to go with a more traditional color scheme, having white as the background and black text with only some styling of the fonts to create contrast. As the black and white concept was a bit dull it was decided that some of the company metrics would be highlighted in color in order to make them visually easier to grasp. As the contrast is higher between the white background and black text this is also easier for the eye.

5.8 Agile Development Process

Most of the development was done according to the agile principle shown in Figure 20, where iterations of code were tested and the final scope of the outcome was altered during the process. The development process was quite good suited for this type of project and it was also easier to get approval on chosen design and features from business representatives when they could test with an actual working site. The drawback with doing things ad hoc on the go was that some of the later development resulted in changes to previously developed parts, mainly due to restrictions on screen size or data presentation issues.

The development process was coordinated with weekly status checks where the latest functionality was presented. The development was split into several smaller tasks, thus making prioritization between tasks and focus areas easier. Also the tasks for the coming week was reviewed every week and decided what to focus on for the coming week.

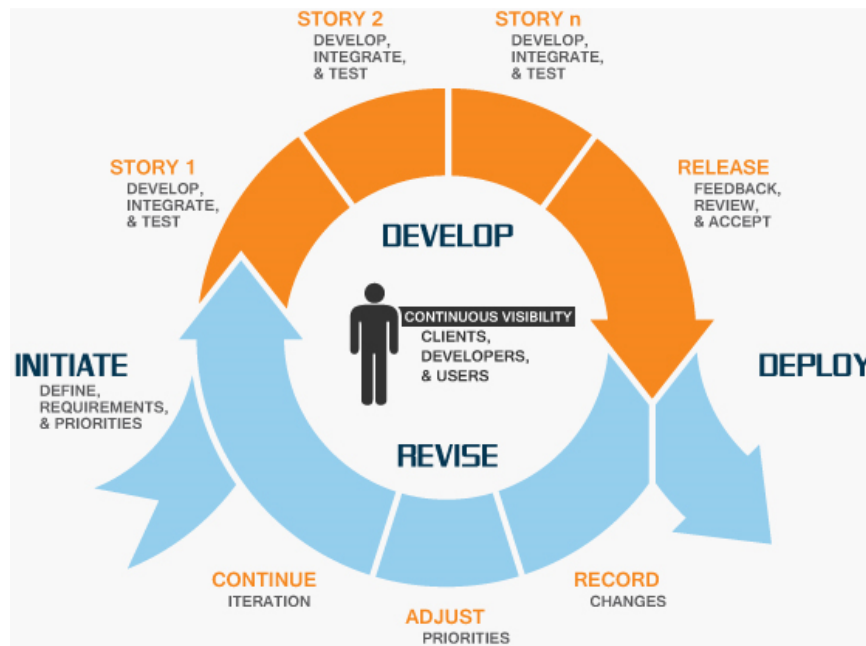


Figure 20. Agile development process [17]

When working according to the agile process the first phase is to initiate the project scope, this is probably the same process for whatever design process is chosen as the framework. The process then continues by developing stories around the business problems, this is to make it easier for business representatives to prioritize and evaluate the different tasks. Depending on the size of the project team one or more stories/tasks can be under development at the same time.

The agile methodology relies on that there is continuous evaluation during the development process which results in a better understanding of all the steps also for the business stakeholders. When the first story is ready for deployment it is still kept in the revision loop, thus continuing the iteration loop also after the traditional development process where deploy means the end of the development phase. Continuous iteration and reviving the scope and stories is probably the most sought after feature within the agile development process, as this enables the project team to adopt quickly to changes in the surrounding settings.

Working according to the agile process enables even a small team to work more dynamically. It is also a good way of getting business representatives active within the development stage, as they can see the progress and influence which parts are delivered at each point in the delivery cycle.

The biggest changes when comparing the agile development process to waterfall methods is according to the Agile Manifesto [18] the following:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

As the development process is not static it enforces an active communication between parties. The more discussions are held regarding the user stories the deeper will the understanding for the underlying problem be also for those people that are only doing the technical part of the project.

5.9 Testing

Most tests were done while developing and thus could not be shared among several testers. Testing of the fully functional module was however done by a small team before deploying the proof of concept solution into live production environment. As the process was done in an agile manner the first iteration of the plugin was put into production tests before all parts were ready, then the second iteration included some new functionality but also bug fixes that had been reported. In total there were 4 major releases and for each of those 1-2 minor bug fixes released.

Testing of the web plugin was divided into three parts: data read/write, data presentation and performance. Performance seemed to be quite good from the beginning, thus that part was not prioritized in the final product. However there was defined a soft metrics for performance: the plugin was not to interfere or indicate that it was in a waiting state when used with live contacts.

Both the read and write tests were divided into two steps, first all SQL queries were run from a specific development tool and in the second step the same queries were tested from within the plugin. This enabled agile methods in adapting the web code to better suite the changes made in SQL queries before and during the development of the container webpage. Also security testing was made on the read and write queries, as to future proof any code that might be reused elsewhere.

The last step was to test the functionality of the user interface and the data presentation. These tests focused on the usability and accuracy of data presented. It was also tested that moving within the plugin was consistent also with keyboard shortcuts, although the majority of usage will be done with a mouse + keyboard combination.

6 Results and Analysis

In this chapter the results of the user interface studies are discussed. The chosen path and color scheme are also described. The targeted hit rate and outcome of the average handling time is described.

6.1 Layout and Constraints

As the plugin developed for the proof of concept had to integrate to the existing environment and applications this set its constraints on the design and development. Although most parts of the plugin development was done within a regular web browser the outcome was quite good suited for the internal web browser of the contact center application, thus minimizing the need for further adaption. The biggest thing to consider when developing for online publishing on a computer screen is to factor the different screen sizes and resolutions in use, also what other applications must be visible at the same time influence the size and layout choices. It was decided that the information should not wrap if the application window was too small, but there was defined a minimum screen size where it should look good when application was used in full window mode. As most of the PC hardware environment used double screens it was agreed that a 19" 3:4 ratio monitor was the minimum viable screen size to test with. Most new screens use the widescreen format, which makes it easier to fit information within a wider view.

When designing the layout of the web plugin and what data to include in which part of the page the usage of the contact center application was monitored in live environment. Based on this information the data was then structured so that the most important parts would most likely always be visible, even if the application window was resized to a smaller size than the minimum supported. By doing this it enabled sales agents to continue with their personal window placement and sizing preferences, but at the same time provide the necessary information within an easily accessible view. There is also the possibility to use the scrollbars to access data that for some reason is outside the initial visible area. When it comes to user experience it is also intuitive to have the relevant data starting from the upper and left side, and then having the more specific information put into the other parts of the page. Another thing that was adopted from the general websites today was the placing of the search-field in the right hand upper corner, as this is probably the most used placing for similar functions on websites today.

To get the overall design working according to set layout constraints CSS is a good standardized way. Using CSS also makes future changes to the look and feel of the web design easier to change when needed. Also color schemes are easily changed if there would be a need to change the containing application or other external constraints put on the developed web plugin.

6.2 Continuity in Workflow

One of the most important things when doing the proof of concept was to reduce the need for manual work and wherever possible speed up the workflow. In order to increase the ease of using the provided information from the recognition plugin into other applications all fields were made easily mark able, as to have easy copy and paste functionality at hand. Also the use of the tabulator button was optimized, as to have faster transition between fields. Other features used to speed up the work process was adding guiding help texts into fields, and also making these automatically clear when the cursor was moved to the field for user input. Some fields were also pre-selected, in order to place frequently used information on the clipboard. Although web techniques give a lot of possibilities for neat features the main concept of the POC was to keep things relatively simple, so that future improvements could be easily adapted without the need of redesigning everything from the beginning. The web code should also where possible be self-explaining in order to make it possible for any web developer to do adjustments in the future with only minimum knowledge of the surrounding applications.

6.3 Hit Rate Improvements

As measurements are a key to finding possibilities for future development the hit rate comparison to the old setup was of most importance. The original input was gathered from only one source market, but the follow up statistics was decided to gather from all countries as it was considered to be more relevant for future improvements. Instead of doing a survey the statistics were gathered directly from page loads, thus giving more accurate results and also making it possible to use continuous tracking without dependencies to other surveys that are running within the application suite. Figure 21

shows the comparison between if the customer was recognized by the automatic process or if correct data was not found.

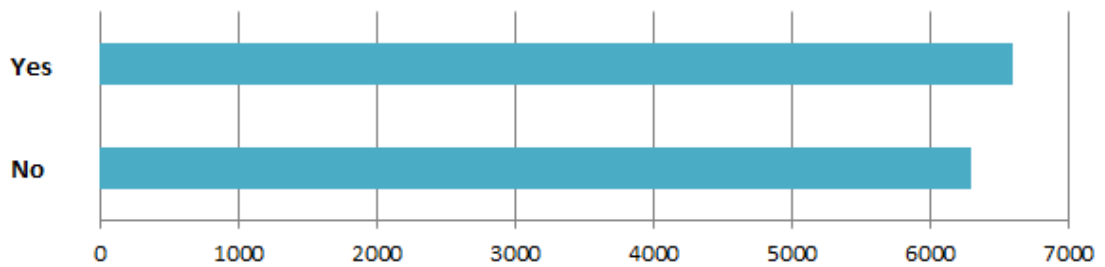


Figure 21. Hit rate results

The original target was set to improve the recognition rate from 40 to 60 percent, but as can be seen from the statistics from 12877 inbound contacts in Figure 21 the successful contacts were only slightly over 52% during the proof on concept. Although the aim of 60% recognition was not met it can still be concluded that the process works better than the old setup. The possibilities for future improvement mainly rely on the underlying customer data and the CRM tools used for data analysis and mining. One other feature that was measured in connection to the hit rate was how many searches yielded results, this was to get an indication if the parameters used for the automatic lookup functionality were performing as expected. From the 6588 contacts that did not give any customer data on the automatic lookup only 1% resulted in a hit when the agent made a manual search. This reinforces the interpretation that the initial lookup functionality viewed from the perspective of the developed plugin is performing as intended and eventual reasons for a lower hit rate that originally estimated should be investigated from a CRM data point of view.

6.4 Impact on Average Handling Time

One of the key figures to analyze was the effect of the improved recognition scenario on the length of calls. Average handling time is one of the most important key performance indicators commonly used within contact centers to measure performance. Figure 22 shows the comparison between the statistics for the year prior to when the new recognition process was implemented. It was chosen to compare statistics for the months instead of just comparing to statistics from the previous 3 months, this was mainly done due to the fact that there are big differences in the call volumes and thus

also the talk times during the year. In this analysis it was only compared contacts in the voice channel, this because e-mail contact handling times might vary much more depending on the type of contacts and thus make comparison very difficult.

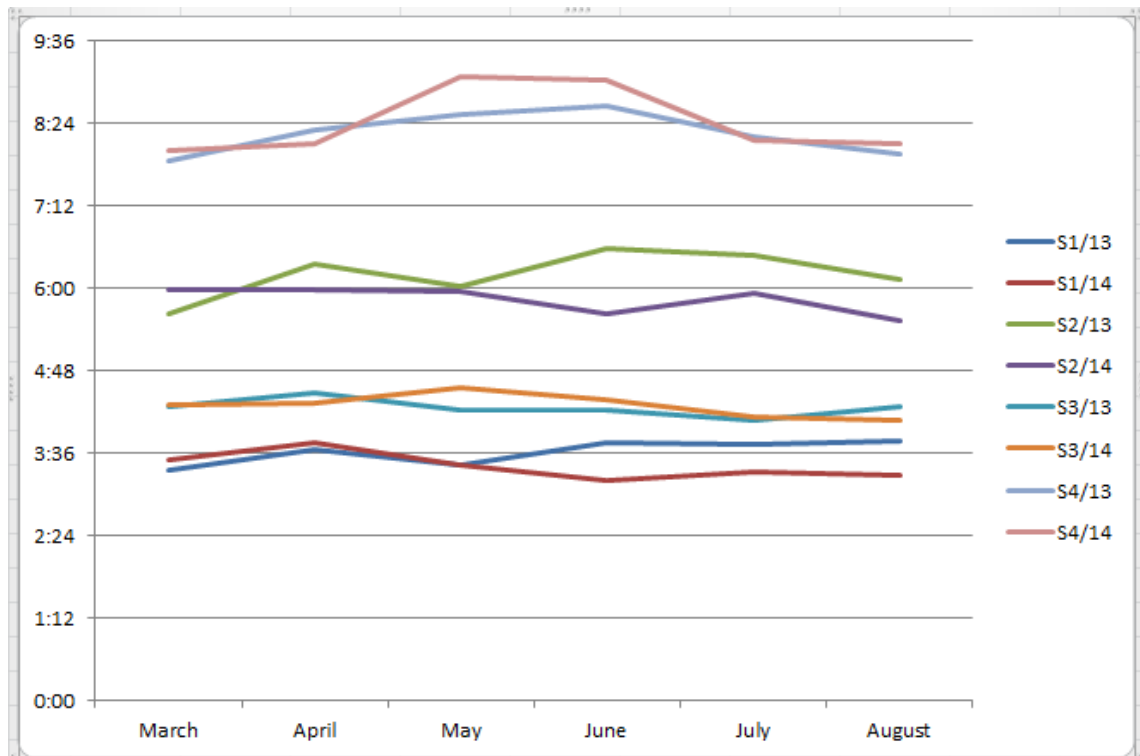


Figure 22. Average talk time comparison

Figure 22 shows a comparison from four different skillsets (S1, S2, S3 and S4). A skill or skillset is the definition in the ACD how and to which agents the contact will be routed, an agent can have one or more skills depending on what type of contacts they are assigned to handle. These skills were chosen to represent different contact types and different source markets. As shown in Figure 22 above, the average talk times vary quite a lot within the chosen timeframe. Figure 22 shows average times for year 2013 and 2014, in order to see the similarity of the overall trend from year to year. However it can be seen that the handling times are shorter after the redesigned recognition process was implemented in the beginning of June 2014. From the graphs in Figure 22 it is also visible that when talk times are getting longer the effect of the recognition process is getting smaller, i.e. as there is a longer time to gather relevant information in between the conversation. In all graphs there is a clear change from May to June when the modification was put into production environment. Some values have clearly gone down after the change (S1 and S2), and others have stabilized the rising trend and

then slowly turned back to shorter times again (S3 and S4). The effect of other simultaneous changes in the working process have not been studied, but as the timeframe analyzed is from a business point of view a time when only minor changes are allowed nothing drastic should have been altered that would then have had an effect on handling times. The chosen skillsets are also from different markets and target groups, thus also minimizing the risk of false data due to too homogenous figures. If a longer timeframe was to be monitored and analyzed the risk of getting incorrect fluctuations also increases as the focus for sales staff differs within the yearly cycle, i.e. there are certain peak times where the routines are adjusted in order to compensate for a higher workload on incoming contacts.

7 Conclusions

This thesis describes the current state of the CRM views in the contact center solution within the reference company TUI Nordic. The analysis was done based on a real world example by re-designing the recognition process and developing a web based plugin for integration to the company's standard contact center application suite. The development was done according to the principles of agile development. There were many minor adjustments to the final outcome during the development process, thus the final product also feels more integrated to the application framework it is running in. The agile process also gives business representatives the possibility to comment and actively be part of the development as they frequently get access to the latest iteration. The downside with not having a set framework is of course that every change needs to be prioritized and evaluated if it is worth the extra time spent on it. However, when looking at the final product it is usually the small changes that are most visible, thus the extra effort spent usually redeems itself in the acceptance of the final product as most issues have already been found and corrected. The agile way of doing development projects depends on good planning and coordination. It is crucial to get business ownership of the development processes, so that there is an understanding of what the resources are allocated to accomplish and also the impact of other ongoing topics that are competing of the same development resources.

The recognition process was redesigned to the extent that is feasible without implementing a lot of new administrative routines. Most of the remaining issues regarding accuracy cannot be solved without looking at the underlying data and the CRM tools. The next step should then be to look at all input data channels and the overall management processes for customer data. The POC showed that it is possible to get quite good results with few improvements. As technology is changing and new systems are invested in recognition is one part of the business processes that needs to be continuously reviewed. The POC showed that current contact channels can easily be integrated, but when new contact channels are designed and implemented also the recognition parameters need to be kept in mind, otherwise there is a risk of losing control of the data and failing to recognize the customers. Also the technical platforms evolve all the time, which is good for performance when the amount of stored data also increases continuously. Based on the statistics gathered and analyzed it is however clear that fine tuning the recognition process can lead to greater optimization of the resources, but foremost to a more efficient customer dialogue and thus happier customers. Alt-

though soft values were not discussed further within the scope of this thesis those cannot be forgotten as they have a great importance in the overall user experience.

In the future it would probably make sense to invest more into the usability of the backend data, i.e. streamlining the access to the information within the CRM system. As custom integrations usually tend to have very specific designs it might also be worth of looking into a new CRM suite with better tools and APIs for integrating third party applications. The obvious next step would be to look at the customer from a new perspective and try to get rid of all excessive middleware where possible, thus linking the UI and data more tightly together and at the same time providing real time updates to data at hand. However as the CRM tools are often quite widespread this thesis will not go further into details regarding recommendations for future development in that area.

Based on the metrics used within this thesis it is quite difficult to find suitable metrics that would not be influenced via other parts of the normal workflow. However, doing a questionnaire with Yes – No answers gives quite reliable answers when it comes to the hit rate. During testing it was also found out that the search-function was used quite widely, thus linking metrics to specific functions within the plugin did not seem reliable enough. It was also decided from a business perspective that continuous follow up and further time invested into setting up metrics was not needed, thus creating of specific scores in the background was decided to be postponed and might be re-evaluated sometime in the future.

The future development potential was also discussed briefly, but as this is highly dependent on the overall working processes and underlying technology this analysis is only valid for the current state. The given guidelines can, however, be referred to also at later stages, but there needs to be a deeper due diligence phase analysis when starting new development projects at a later stage to validate the desired outcome and the relevance of earlier described scenarios. With the constraints in mind the overall process described in this report should be viable also within different market segments and industries, thus making this proof of concept neutral to industry or other dependencies.

References

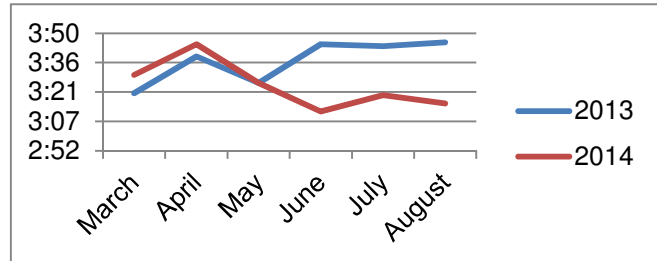
- 1 TUI Group. TUI Group presentation [Internet]. 2015 [cited 12.9.2015]. Available from: <https://www.tuigroup.com/en-en/investors/reports-and-presentations>
- 2 Gartner. Magic Quadrant for Contact Center Infrastructure, Worldwide. 2015.
- 3 Peppers & Rogers Group. Optimizing Business Productivity: Do More for Your Customers and Your Business [Internet]. 2011 [cited 22.5.2014]. Available from: http://az26122.vo.msecnd.net/docs/WPMicrosoft_UShigh080411.pdf
- 4 Keightley N. and Lai E. Managing the Customer Experience: How to Maximize the Lifetime Value of Your Most Precious Asset. Avaya Inc; 2013.
- 5 Peppers, Don, and Rogers, Martha. Managing Customer Relationships : A Strategic Framework (2nd Edition). Hoboken, NJ, USA: John Wiley & Sons, 2011.
- 6 Avaya Inc. 2016. Available from: <http://www.avaya.com>
- 7 Telecomunicaciones XSpand. 2015 [cited 10.1.2015]. Available from: <http://www.xspandllc.com/site/images/productos/avaya/callcenter/auracontact.jpg>
- 8 Compare Business Products. 2015 [cited 10.1.2016]. Available from: <http://www.comparebusinessproducts.com/resources/item/contact-center-comparison-chart>
- 9 Genesys. Best Practices for a Seamless Omnichannel Customer Experience. 2015.
- 10 Webopedia. 2015. Available from: <http://www.webopedia.com/TERM/A/asynchronous.html>
- 11 NetLingo. 2015. <http://www.netlingo.com/>
- 12 CloudTags. 2016. Available from: <http://omnichannel.me/what-is-omnichannel/>
- 13 Kriegel A. Discovering SQL: A Hands-on Guide for Beginners. Wrox; 2011.
- 14 Shah R. and Thaker B. Microsoft SQL Server 2012 Performance Tuning Cookbook. Packt Publishing Ltd; 2012.
- 15 Hall, Richard H, Hanna, Patrick. The impact of web page text-background colour combinations on readability, retention, aesthetics and behavioural intention. Behaviour and Information Technology, May-June 2004, vol. 23, no. 3. PDF.
- 16 Hill, Alyson L. Readability of Websites With Various Foreground/Background Color Combinations, Font Types And Word Styles. Department of Psychology, Stephen F. Austin State University [Internet]. 2012 [cited 8.9.2015]. Available from: <http://www.mmeissner.de/AHNCUR.html>
- 17 AGILE Development – Modern Software Development Strategy. CompSciwonders [blog on the Internet]; 2014 [cited 4.8.2015]. Available from:

<https://compsciwonders.wordpress.com/2014/09/09/agile-modern-software-development-strategy/>

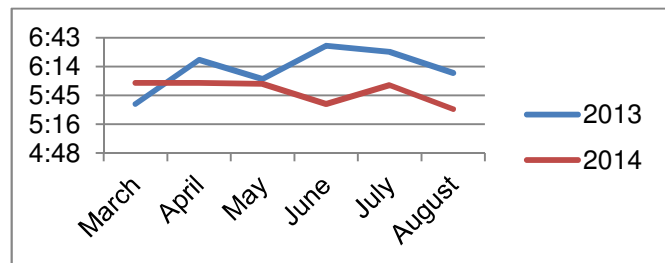
- 18 Agile Manifesto. 2001 [cited 5.8.2015]. Available from: <http://agilemanifesto.org/>

Statistics Overview

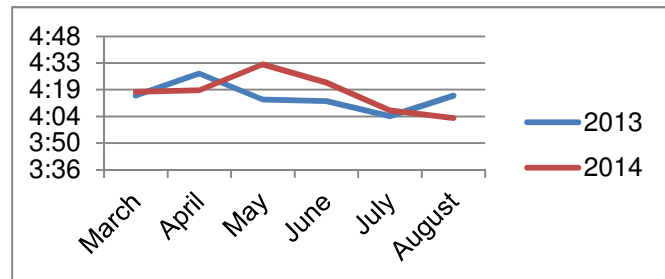
Skill 1	2013	2014
March	3:21	3:30
April	3:39	3:45
May	3:26	3:26
June	3:45	3:12
July	3:44	3:20
August	3:46	3:16



Skill 2	2013	2014
March	5:37	5:58
April	6:21	5:58
May	6:02	5:57
June	6:35	5:37
July	6:29	5:56
August	6:08	5:32



Skill 3	2013	2014
March	4:16	4:18
April	4:28	4:19
May	4:14	4:33
June	4:13	4:23
July	4:05	4:08
August	4:16	4:04



Skill 4	2013	2014
March	7:51	8:01
April	8:18	8:07
May	8:32	9:05
June	8:39	9:02
July	8:12	8:09
August	7:58	8:07

