CRM System and Consolidated Information Resource: Ways of Creation and Interaction

Oleksandr Markovets, Roman Holoshchuk, Svitlana Holoshchuk and Nestor Dumanskyi

Lviv Polytechnic National University, 12 Bandera street, Lviv, 79013, Ukraine

Abstract

This work defines general concepts of the system of customer relationship management (CRM), the relevance of the researched question in the analysis of functionality of CRM-systems and the main objectives to be achieved in the process of the work. While performing it we achieved our goal and developed a mobile interface with an access to CRM-resource. The article describes the algorithms, which are implemented in the process of developing a mobile application, especially the algorithm of the authorization/authentication and the algorithm of performing any query to the back-end of the CRM-system. The scheme of data streams exchange is presented in details. It also concentrates on the ways of high-level architecture of iOS-application and the results of the completed work.

Keywords 1

Mobile interface, CRM-systems, Model-View-Controller (MVC), iOS-application, Xcode, SalesShtorm

1. Introduction

Modern enterprises which carry out any type of activity possess their own database that contains information on the company's activities, cooperation with other companies, suppliers, customers, records about them, history of cooperation, as well as personnel of the enterprise, etc. The existence of such a database is an essential and constituent part of the enterprise development and success. To arrange and keep records of any piece of documentation, to change any data in these databases, or to search for the necessary information, the company needs to use special software automation (automated systems) [1]. It is a system which collects, stores and processes data and significantly simplifies the interaction with the database. It gives a solution to a much better level of enterprise performance and its cooperation with other enterprises, improves the level of customer service, assists in arranging documentation, making reports and much more. Most companies/enterprises have a customer-orientated policy; thus an effective customer relationship management receives most significant consideration for further successful development. Systems of this type become more widespread nowadays [2].

The relevance of the research article is that despite the fact that business automation is rather well spread, there are a lot of companies that still work with a large amount of data and have difficulties in processing it with the help of modern office applications (which might be rather obscure), or modern systems which do not correspond to the goals and objectives of the company. It results in an aroused demand to automate the workflow which has to be accustomed to the user needs. Modern representation of such a system should possess not only a web-presentation but also a mobile interface which gives an employee a possibility not to stick to the working place and computer but to have a possibility to be flexible and work remotely while having a mobile access to the database. The

1COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: oleksandr.v.markovets@lpnu.ua (O. Markovets); roman.o.holoshchuk@lpnu.ua (R. Holoshchuk); svitlana.l.holoshchuk@lpnu.ua (S. Holoshchuk); nestor.o.dumanskyi@lpnu.ua (N. Dumanskyi)
ORCID: 0000-0001-8737-5929 (O. Markovets); 0000-0002-1811-3025 (R. Holoshchuk); 0000-0001-9621-9688 (S. Holoshchuk); 0000-0001-6908-2751 (N. Dumanskyi)

development of mobile interfaces is also important in various areas of economics as it provides an access to specialized information systems [3].

The aim of this research is to describe the stages of developing a mobile application interface with an access to the consolidated information resource of customer relationship management (CRM-system), which should be convenient and easy to use [4].

For this reason, we should accomplish the following tasks:

- to study and review the works in this feld;
- to define a problem and the stages of its solution;
- to find methods and ways of solving the problem;
- to develop a mobile interface of SalesShtorm CRM-system.

2. Models and Technologies of Customer Relationship Management

Customer relationship management system (CRM) is a software application designed for business to automate customer interaction strategies. Its aim is to increase sales, optimize marketing and improve customer service by storing information about customers and their cooperation, establishing and improving the quality of business procedures and further consideration of the outcomes [5].

Customer relationship management is a model of interaction that determines that the center of the entire business philosophy is a client, and its main tasks include actions to support effective marketing, sales and customer service [6]. Implementation of these business goals includes collection, storage and analysis of information about consumers, suppliers, partners, as well as the data on the company's internal functions. Functionality to support these business goals is developed via sales management, marketing management, customer service management, and call centers [7].

There are two main types for viewing the information provided by the CRM system. The first one is a desktop application with the CRM programme installed on the client's computer i.e. special software without which the CRM does not work [8].

This solution has its own advantages and disadvantages. It is believed that the installation of the programme on the client side provides a high level of data protection. If the company is seriously concerned about the protection of transmitted data, it is better to choose a system with software installation [9].

The second type of access point is called online application. In this case the application could be found on the internet and there is no need to install it on the client computer. It is a convenient type to use as it provides the ability to quickly access CRM system from different machines and satisfies business flow requirements [10, 11].

CRM systems can also be subdivided into two types according to the model of distribution to consumers – SAAS and Standalone. Together with the boosting demand in CRM systems, the need for mobile CRM technologies also increases.

In a conceptual framework, CRM applications is a measure of CRM functionality. Also, the possible positive link between the CRM functionality, i.e., the extent and variety of CRM software applications used in an organization and the performance outcomes [12-15].

Mobile version of CRM system allows customer service representatives or sales representatives to handle necessary and useful information about the customer which helps them to solve business issues quickly and efficiently. Having a mobile application, they are able to check information about the customer with whom they work or a potential customer with whom they plan to agree a partnership. Moreover, mobile CRM allows to view not only transactions and stages of cooperation in which the manager is but also an updated list of company's products and promotions. When the company's sales manager using the mobile CRM system in the online mode, without being limited by the office, is able to assist the customer, it will increase the loyalty of the latter and the chance to repurchase the product or service [16, 18].

We would like to stress that the special software for mobile CRM allows employees to transfer to mobile devices only the data they need without overloading the system with the desktop computer functions. Every employee should navigate the data displayed on the screen of the mobile device with ease and convenience. Mobile CRM should work smoothly with the possibility to find the most necessary information very quickly and with a minimum number of operations.

CRM system is a modern approach to the concept of consolidated information resource interaction with customers. CRM strategy determines the relationship with customers on various business stages – it includes advertising, sales, delivery and customer service, design and production of new products, service delivery, invoicing, etc. Automation of these processes is a significant technical contribution for companies that want to achieve a successful interaction with customers. The aim of this strategy is to improve the quality of customer service which, in its turn, enables to increase the company's market share and results in its profitability.

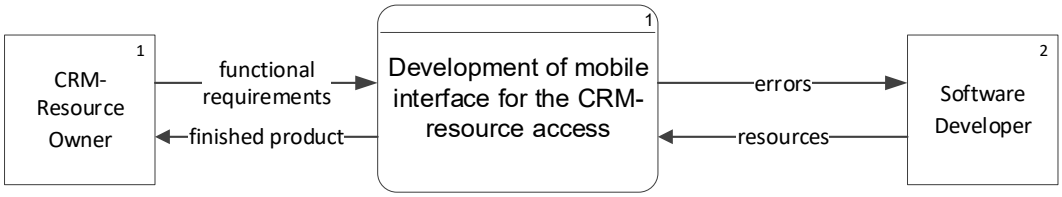

Figure 1: Context diagram of data flows

A context diagram of data flows presents the main purpose of our work – to develop a mobile interface with an access to the CRM-system and it is shown in Figure 1 The diagram demonstrates its two constituent parts – CRM-Resource Owner and Software Developer. The owner sets functional requirements for the future mobile application and the CRM resource itself, which should be accessed through the mobile interface. The programmer is a key figure in the development of mobile application software. His task is to receive requirements and resource, to carry out the main part of the work by applying his skills, knowledge and experience.

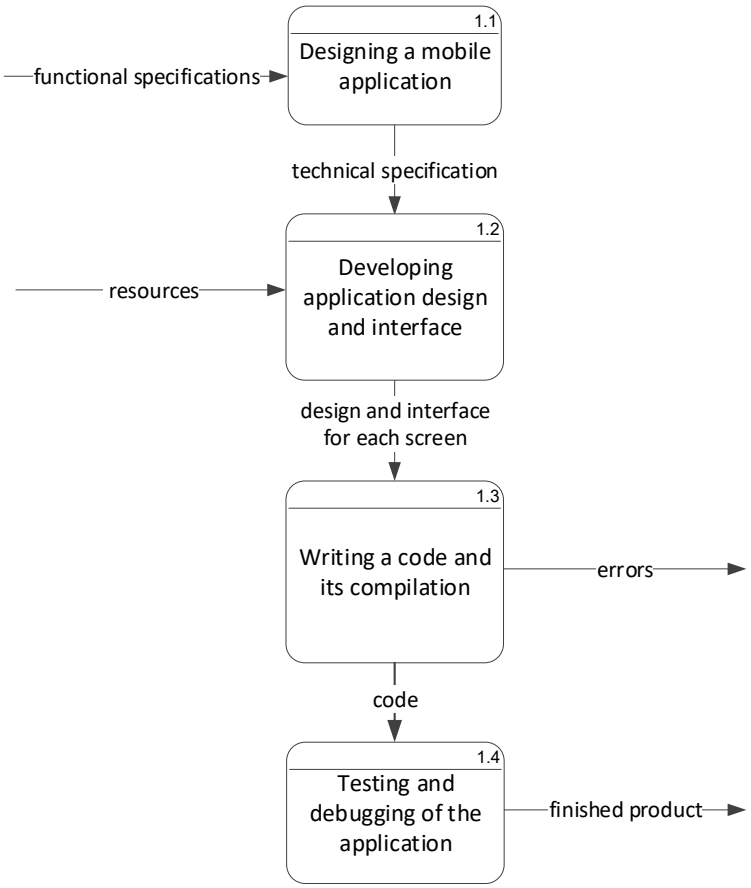

Figure 2: First level of detailization

The first level of the diagram of detailization is presented in Figure 2. It shows the main stages of development of the mobile application. At the first stage, the mobile application is designed. It includes the following tasks: to form the technical specifications, to process the future functionality of the

programme, to create the special map of interaction between screens, and to choose algorithms of performance of the technical task. The input parameters are the functional requirements for the future mobile application.

At the stage of the application design and interface development, each of the screens is tested, the layout for each of them is elaborated and it is decided which buttons, pictures, icons and functionality to place there. The input data for this stage are resources which include icons, pictures, access to CRM-resource, etc.

At the stage of writing a code and compilation, a programmer develops an application by writing its code. This is where the connection between the developed application and the electronic CRM database is implemented. The process of code writing requires frequent compilation to verify the operation of each piece of code. Compilation allows timely detection of errors in the code and their processing/elimination.

At the stage of application testing and debugging, its finished version is checked and found defects and inaccuracies of its work and presentation are corrected. The completion of this stage results in the finished product.

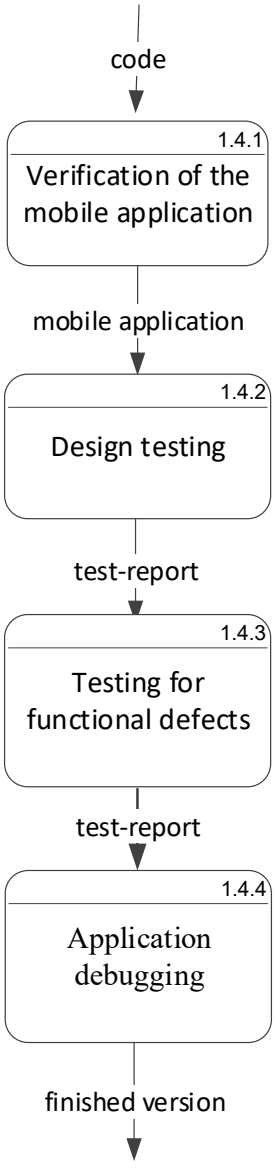

Figure 3: Second level of detailization

Detailization of the testing process and debugging of the application is shown in Figure 3. When creation of the application is completed, there is a need to test it. Testing includes two stages – design testing and testing for functional defects. Functional defect testing reveals inaccuracies that need to be debugged for proper functionality and make the finished application easy to use. Then it becomes available for the customers in online stores such as PlayMarket, AppStore or Ovi Store. The choice of store depends on the target audience and the technology it uses.

The database of this CRM-resource also contains 36 additional entities that provide the ability to detail basic data, processes and more. In this database management system (DBMS), tables are tied by logical connections. In Figure 4, the connections between the tables are represented.

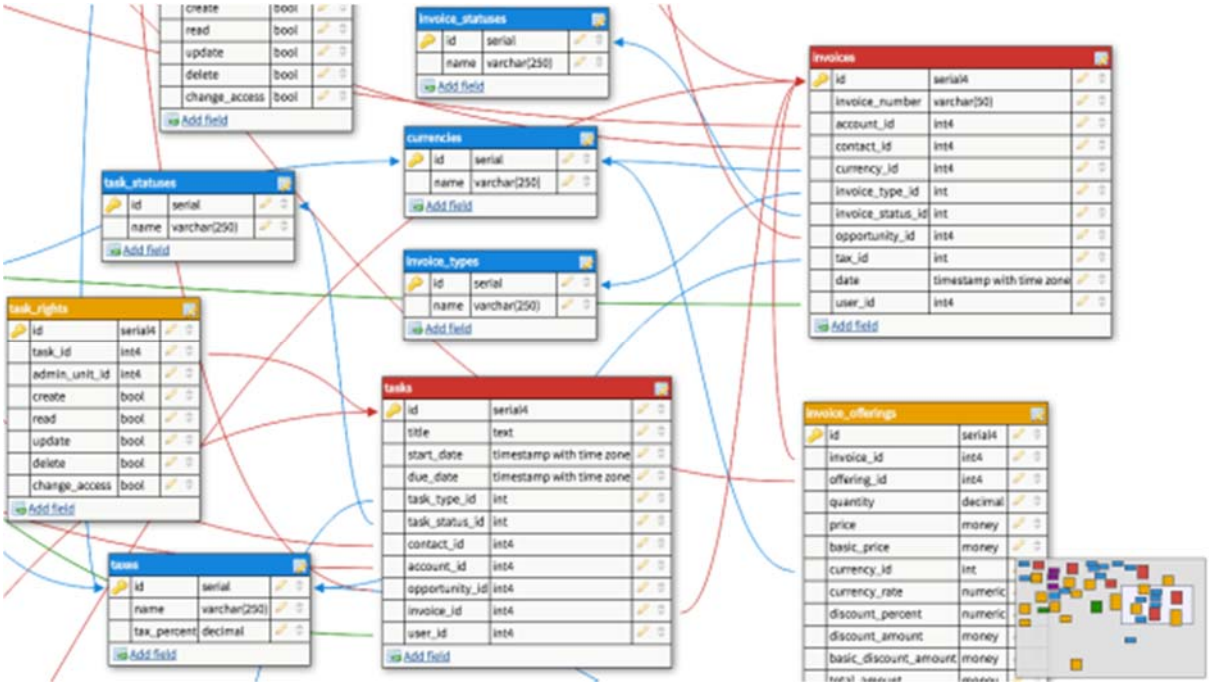

Figure 4: Connections between the tables in DBMS (PostgreSQL)

The expected outcomes of using CRM-resource are the ability to automate and improve business processes in such areas as customer service and support, sales, marketing, as well as management and analysis.

Consolidated information resource accumulates and stores important company data in an organized system.

3. Methods and means of creating a mobile interface for access to CRM-resource

To accomplish this task, it is necessary to work out the following methods to solve the problem: methodology of application creation – the life cycle of its creation, client-server architecture, which is the basis for communication of client applications with databases stored on a remote server, and MVC template.

The application lifecycle is divided into several main stages. They include application development, design and interface elaboration, code creation and compilation, testing of the application versions, application debugging and its preparation for implementation.

Client-server architecture (Figure 5) is widely used to interact with the server in order to work with shared data from various applications on the network. It provides the following main components:

- a set of servers that provide information or other services to programs that access them;
- a set of clients who use the services provided by servers;
- a network that provides interaction between clients and servers.

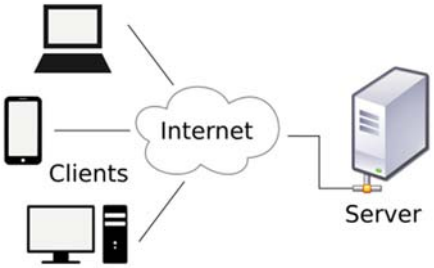

Figure 5: Client-server architecture

Model-View-Controller (MVC) is also an important architectural template for software design and development [19].

The MVC architectural template divides the programme into three parts (Figure 6). In the triad, the tasks of the Model component include to store data and provide an interface to them. The task of the View component is to display data to a user. The task of the Controller component is to manage its constituents, receives signals in response to user actions, and reports changes in Model component. Thus, the components of the template are presented as separate functional parts with a certain responsibility distributed between them. In the MVC template, Model and View components do not interact directly, the interaction between them is provided by the Controller one. It means that MVC aims to separate business logic from the user interface.

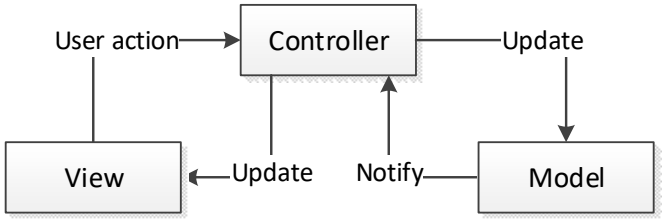

Figure 6: MVC template

As a modern programming language, Swift includes many features and capabilities, incorporates useful ones from other languages (including Objective-C) and ignores unnecessary ones.

Consdering the analysis conducted above we would choose Swift programming language for our further investigation.

After carrying out a system analysis, it is determined that the development of the iOS application will use the object-oriented programming approach, the Swift programming language and the Xcode integrated software development environment [20-22].

4. Implementation of the mobile interface development process for access to the CRM-resource and the results obtained

For practical implementation of the process of mobile interface development, the SalesShtorm CRM system is chosen – a rather new system in the modern market of CRM-systems. The system is represented by a web application that contains the most necessary functionality for using the CRM system. Its sections include analytics, tasks, contractors, contacts, accounts, sales, products, users, administration and directories. The application provides an intuitive and user-friendly interface for using the CRM system.

As it was mentioned above, the development of a mobile application for the iOS platform is decided to implement. It should be stressed that the data flow between the web and iOS versions is not fundamentally different.

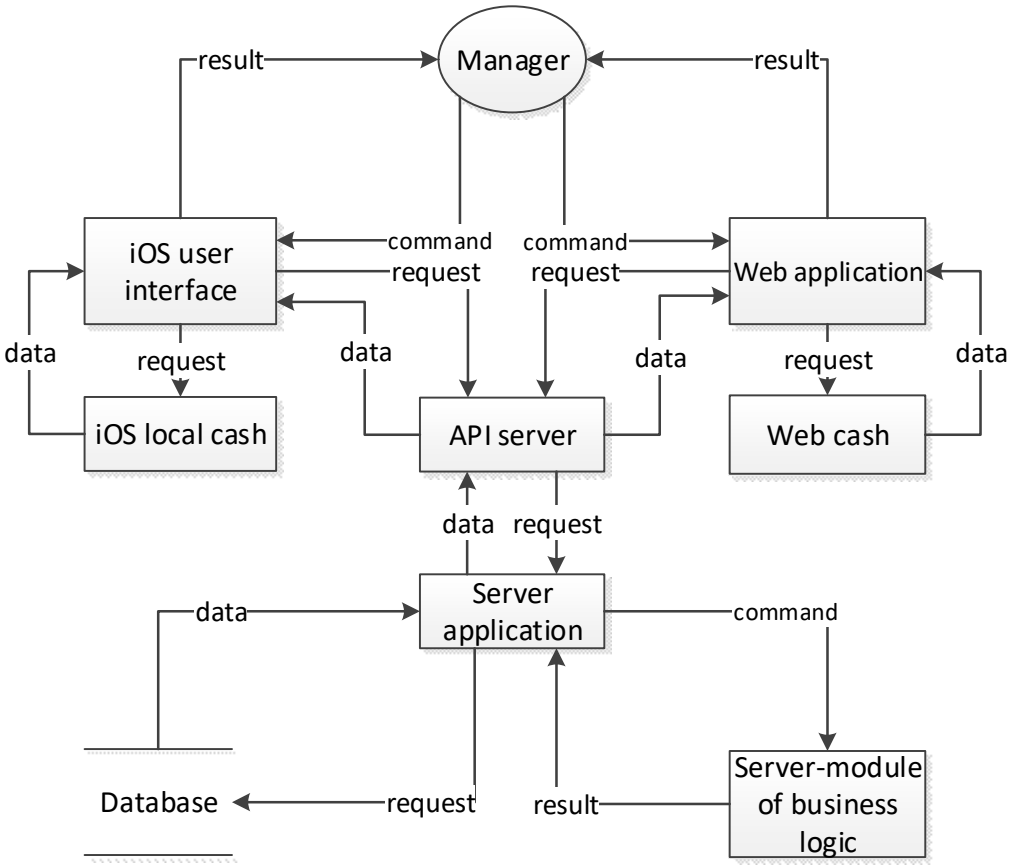

Figure 7: Data flow exchange in CRM system

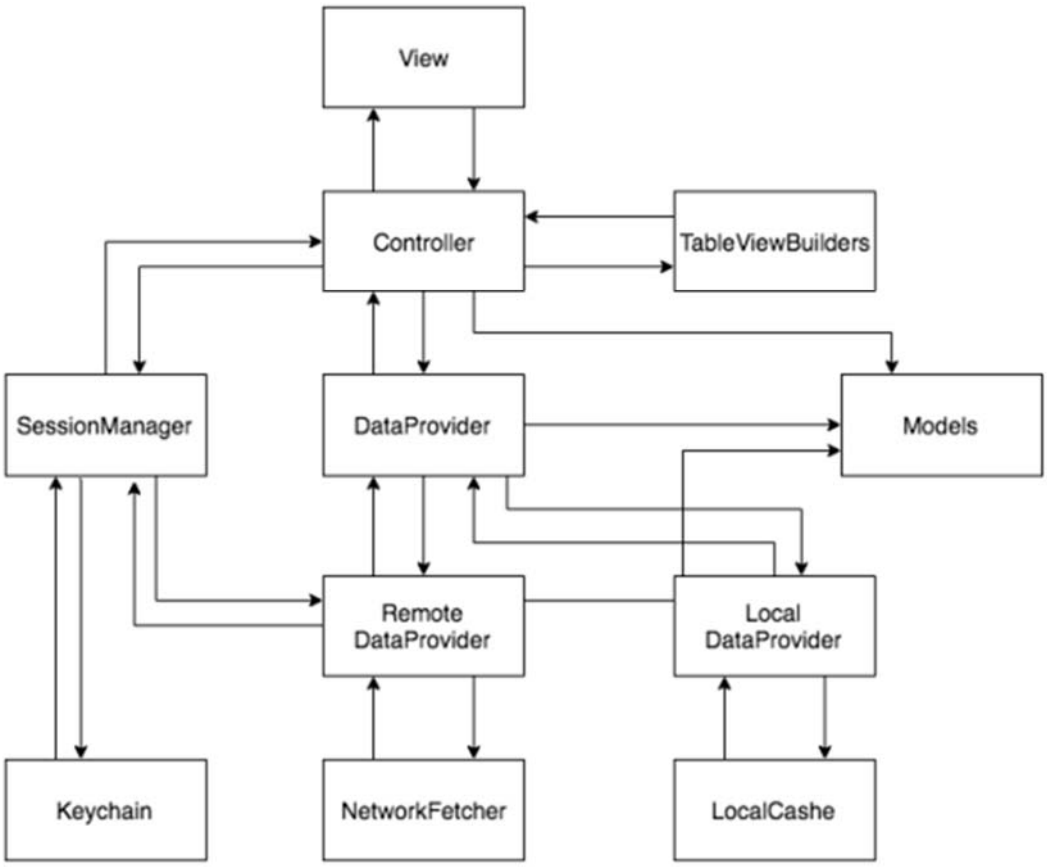

Figure 8: High-level architecture of the iOS application

The flow of data from the user to the system is demonstrated in Figure 7. The manager using the interface of the iOS application creates a lot of instructions while working. Each of them has to be processed by iOS-application and decides where to draw the data. The iOS application can take data from the local cache which is created and accumulated as the application operates, and cleared when the application is closed. The cache can also be cleared if there is not enough RAM. If no data is found in the local cache, the iOS application accesses the CRM server application via the API (Figure 8). The server application, in its turn, communicates with the business logic module. This module, for example, can decide whether a user has the appropriate access rights to perform a particular request/command and retrieve data from the database. In other words, if there is a request from the iOS application to the server part of the CRM system, the error occurs. If the business logic module accepts the corresponding request, the server addresses a database to receive data. Data received from a database is returned in reverse order. Databases return information to the server part of the CRM system, which in turn gives an asynchronous response to the data or the command result.

More detailed description of the high-level architecture of the iOS application is presented below.

When opening the iOS application, the user first sees the authorization page (Figure 9). Here you can enter a subdomain, login and password. The subdomain determines which part of the system to connect. Each company has its own sample of the system, and, correspondingly, a unique link.

Figure 9: Authorization page

Then the user moves to the opening page of all tasks (Figure 10). They are viewed in chronological order with the new ones to appear first. Simultaneously, there is a transition to a detailed presentation of the task and addition of a new one. It also provides the following functions: to filter a present task and search by a task name. If there is an urgent need to change the task status, it is enough to drag it to the left.

The system provides three types of tasks:

- New task,
- Tasks in the process of execution,
- Completed (implemented) task,
- Completed (not in time) task.

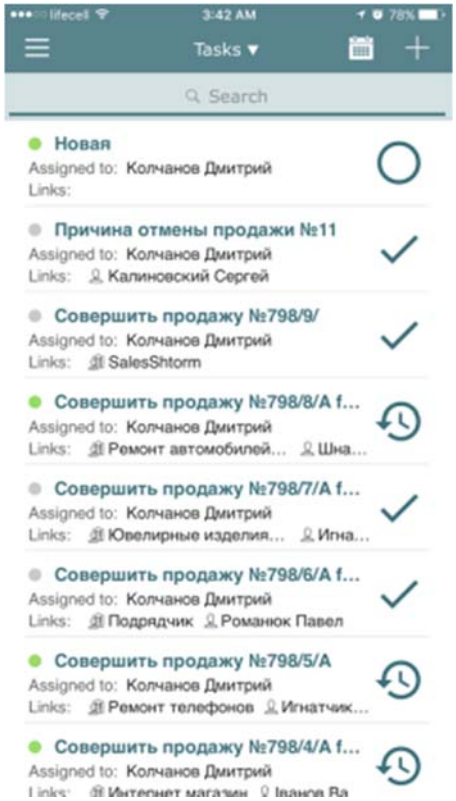

Figure 10: List of tasks

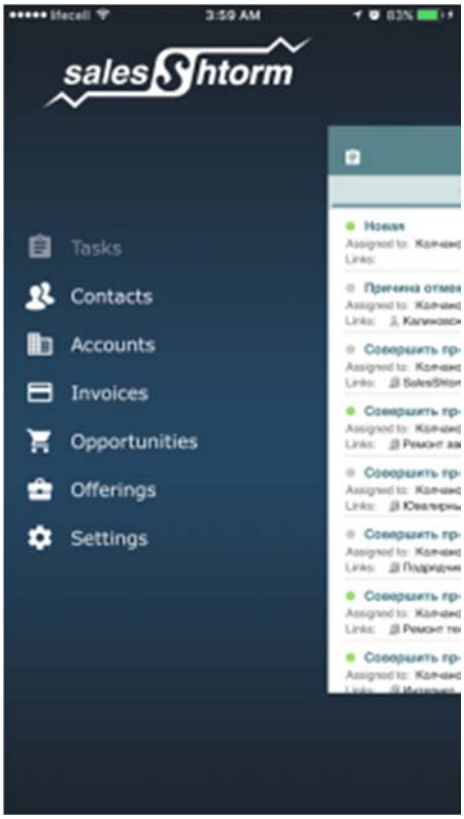

Figure 11: Navigation panel

To navigate between the system sections there is a navigation panel on the left (Figure 11). It is displayed when the user presses a menu button of the panel on the left which enables a quick switching between the system parts.

5. Conclusions

After a detailed analysis of investigations in the field of CRM-systems, the main principles of its work are studied. It is identified and described the role of CRM-systems in terms of the main company goal while applying CRM-systems into their work – to develop long-term and loyal relationships with customers. The architecture of the system and the outcomes of its integration into the company's work are considered.

The process of creating a mobile interface for access to CRM-resource is visually presented via data flow diagrams, and first and second levels of detailization.

The main algorithms of work with CRM-resource, as well as the main methods and means of implementing the solution are considered. It is agreed to apply an object-oriented programming approach, the Swift programming language and the integrated software development environment (Xcode) while creating the iOS application.

In the process of carrying out this work, we achieved our aim and the mobile interface to access the CRM-resource is created. The algorithms implemented in the process of mobile application development are described, namely, the authorization/authentication algorithm and the algorithm for sending any request to the server part of the CRM system, the scheme of data flow exchange is given. The high-level architecture of the iOS application and the results of the completed task are also presented.

6. References

[1] M. Khorraminia, Z. Lesani, M. Ghasvari, L. Rajabion, M. Darbandi and A. Hassani, A model for assessing the impact of cloud computing on the success of customer relationship management systems (case study: agricultural companies). Digital Policy, Regulation and Governance, (2019). doi:10.1108/DPRG-03-2019-0016.

[2] M. Pour, F. Mamani and M. Rahimzadeh, How Customer Relationship Management (CRM) and Innovation Influence Business Performance Mediating Role of Innovation. International Journal of Customer Relationship Marketing and Management, Volume 9(2), (2018), 1-15. doi:10.4018/IJCRMM.2018040101.

[3] S. Makara, L. Chyrun, Y. Burov, Z. Rybchak, I. Peleshchak, R. Peleshchak, R. Holoshchuk, S. Kubinska, A. Dmytriv, An Intelligent System for Generating EndUser Symptom Recommendations Based on Machine Learning Technology. In: Computational Linguistics and Intelligent Systems, COLINS, CEUR workshop proceedings, Volume 2604, (2020), 844-883.

[4] S. Deng and F. Zhang, Design of a knowledge-based e-CRM system: A case of freight forwarding industry, (2008), doi: 10.1109/WiCom.2008.2521.

[5] K. Saeed, V. Grover, W. Kettinger and S. Guha, The successful implementation of customer relationship management (CRM) system projects. ACM SIGMIS Database: the Database for Advances in Information Systems, Volume 42(2), (2011), 9-31. doi: 10.1145/1989098.1989100.

[6] Z. ZHAO and H. LI, Critical Success Factors of CRM System Implementation. DEStech Transactions on Social Science, Education and Human Science, (2017). doi:10.12783/dtssehs/mess2016/9738.

[7] M. Pohludka and H. Štverková, The Best Practice of CRM Implementation for Small- and Medium-Sized Enterprises, Administrative Sciences, Volume 9(1), (2019). doi:10.3390/admsci9010022.

[8] E. Taylor, The Future of Marketing Science. Research World, Volume 2017(63), (2017), 57-57. doi:10.1002/rwm3.20505.

[9] D. Puligheddu, R. Tonelli and M. Marchesi, Managing CRM with Fabric Hyperledger blockchain technology. Blockchain and Web 3.0, (2019), 223-239. doi:10.4324/9780429029530-14.

[10] O. Markovets, R. Pazderska, N. Dumanskyi, I. Dronyuk, Analysis of citizens' appeals in heterogeneous web services. CEUR Workshop Proceedings, Volume 2392, (2019), 184-198.

[11] B. Khashab, S. Gulliver, and R. M. Ayoubi, Scoping and aligning CRM strategy in higher education institutions: practical steps, J. Strateg. Mark., (2020), doi: 10.1080/0965254X.2020.1823458.
[12] J. Boarim and A. R. Cavalcanti Da Rocha, Quality Characteristics of CRM Systems, (2020), doi: 10.1145/3439961.3439980.
[13] E. V. Cherkasova and M. R. Zainullina, Digital assistants in managing customer relationships in modern companies, vol. 139. (2021).
[14] W. John, Modelling CRM applications and its impact on relationship outcomes: a B2B relationship perspective. International Journal of Electronic Customer Relationship Management, Volume 11(4), (2018), 347. doi:10.1504/IJECRM.2018.096240.
[15] O. Markovets, N. Vovk, Y. Turchyn, O. Smirnov, Model of informational support for social network administrators' content creation. CEUR Workshop Proceedings, Volume 2616, (2020), 125-136.
[16] G. Krasner and S. Pope, A cookbook for using the Model-View-Controller user interface paradigm in Smalltalk-80. Journal of Object-Oriented Programming, Volume 1(3), (1988), pp.26-49.
[17] A. Faed, A. Ashouri, and C. Wu, The efficient bond among mobile commerce, CRM and E-loyalty to maximise the productivity of companies, Proceedings - 3rd International Conference on Information Sciences and Interaction Sciences, (2010), pp. 312–317, doi: 10.1109/ICICIS.2010.5534812.
[18] H. Chavan and S. Sane, Mobile database cache replacement policies: LRU and PPRRP, vol. 131 CCIS, no. PART 1, (2011).
[19] J. Dörndorfer, C. Seel, and D. Hilpoltsteiner, SenSoMod-A modeling language for context-aware mobile applications, in MKWI 2018 - Multikonferenz Wirtschaftsinformatik, vol. 2018-March, (2018), pp. 1435–1446.
[20] V. Savchuk, Y. Vykyuk, V. Pasichnyk, R. Holoshchuk and N. Kunanets, The Architecture of Mobile Information System for Providing Safety Recommendations During the Trip, Volume 938, (2020).
[21] V. Geist, M. Moser, J. Pichler, R. Santos, and V. Wieser, Leveraging machine learning for software redocumentation - A comprehensive comparison of methods in practice, Softw. - Pract. Exp., vol. 51, no. 4, pp. 798–823, (2021), doi: 10.1002/spe.2933.
[22] R. S. Dremin et al., Remote monitoring software for induction soldering installation, IOP Conference Series: Materials Science and Engineering, vol. 734, no. 1, (2020), doi: 10.1088/1757-899X/734/1/012134.