

И. Е. Букшев

*Новосибирский государственный университет
ул. Пирогова, 1, Новосибирск, 630090, Россия*

*Центр финансовых технологий
ул. Шатурская, 2, Новосибирск, 630055, Россия*

bukshev@gmail.com

MEDILUX – СЕРВИС ИНТЕЛЛЕКТУАЛЬНОГО ФОРМИРОВАНИЯ РАСПИСАНИЯ ПОСЕЩЕНИЙ МЕДИЦИНСКИХ УЧРЕЖДЕНИЙ

Предложен новый вид обслуживания – «интеллектуальная регистратура»: в мобильном приложении обеспечена возможность выбора сотрудников медицинского учреждения, далее происходит анализ графика работы выбранного врача и свободного времени пользователя при помощи алгоритмов программирования в ограничениях с целью определения наилучшего времени записи на прием. База знаний о свободном времени пользователя формируется на основе задач, которые агрегируются с мобильного устройства и популярных сервисов по управлению задачами. Для случая, если пользователь не знает, к кому обратиться, продукт снабжен «умным» чатом, в котором можно описать проблему. Текст отправится на сервер, где произойдет синтаксический разбор и семантическое сопоставление с конкретной специальностью врача. В базе данных хранится информация обо всех посещениях и врачебных выписках (электронная медицинская карта), что позволяет, например, напоминать пользователю о необходимости принятия медикаментов.

Практическая ценность продукта заключается в автоматизации бизнес-процесса «прием пациентов», что приводит к экономии времени пациентов, обеспечению высокой доступности услуг и оптимизации трудовых затрат в медучреждениях.

Ключевые слова: регистратура, программирование, мобильное, приложение, чат, запись, прием.

Актуальность

В настоящее время проблема качества медицинского обслуживания стала предметом внимания властей и средств массовой информации, поскольку уровень развития здравоохранения говорит о развитии государства в целом. Бизнес-процесс «прием пациентов» является отличным примером, так как пока еще посещение поликлиники требует много усилий и связано с негативными последствиями, такими как:

- 1) жалобы пациентов из-за недоступности медицинской помощи;
- 2) перекрестное инфицирование из-за скопления пациентов в одном месте;
- 3) самолечение и, как следствие, увеличение материальных затрат на лечение пациентов с осложнениями;
- 4) периодические стрессовые ситуации, связанные с работой, у персонала медицинских учреждений;
- 5) падение рейтинга и имиджа, что является существенной проблемой для негосударственных медицинских учреждений.

Анализ возможных источников проблемы дал понять, что к вопросу следует подходить с нескольких сторон. Возможные факторы и причины:

Букшев И. Е. Medilux – сервис интеллектуального формирования расписания посещений медицинских учреждений // Вестн. НГУ. Серия: Информационные технологии. 2018. Т. 16, № 2. С. 41–48.

- 1) кадровая проблема – отсутствие должного количества врачей;
- 2) нежелание врачей работать в государственных медицинских учреждениях;
- 3) отсутствие должного взаимодействия между подразделениями медицинской организации и четких алгоритмов работы каждого подразделения;
- 4) ограничение информированности населения о возможности получения медицинской помощи, несвоевременные обращения, необходимость для пациентов в «лишних посещениях» (запись, чтобы «узнать» или «спросить»);
- 5) избыточные временные затраты на оформление медицинских документов, дублирование информации, растущая отчетность.

Исходя из перечисленных проблем можно сделать вывод, что не существует универсального решения даже для такого, казалось бы, малого звена в функционировании медицинского учреждения, как бизнес-процесс «прием пациентов».

Предлагаемое решение

Частично помочь пациентам может «клиент-серверное» решение, где в качестве «клиента» будет выступать мобильное приложение, предоставляющее пользователям электронную медицинскую карту, электронную регистратуру и электронный чат с врачами. Данное решение не ново, но предлагаемый сервис отличается от прочих тем, что способен *автоматически определять* удобные дату и время приема как для конечного пользователя, так и для лечащего врача. Помимо этого, в сервисе предлагается интеллектуальный чат, в котором пользователь сможет получить общую информацию или советы в режиме реального времени.

Таким образом, сервис нацелен на анализ повседневного ритма и рабочего графика с последующей возможностью *интеллектуального планирования* и *автоматического осуществления записи* в медицинские учреждения. Мобильное приложение будет выступать интеллектуальным звеном-помощником между пациентами и поликлиниками.

Научная новизна

В работе предложен новый вид обслуживания – «интеллектуальный прием пациентов»: в мобильном приложении имеется возможность выбора сотрудников медицинского учреждения, далее происходит анализ графика работы выбранного врача и свободного времени пользователя при помощи алгоритмов «программирования в ограничениях» с целью определения наилучшего времени записи на прием. База знаний о свободном времени пользователя формируется на основе задач, которые агрегируются с мобильного устройства и популярных сервисов по управлению задачами (Trello, Google Tasks, Wunderlist, Apple Reminders и др.). Для случая, если пользователь не знает, к кому обратиться, продукт снабжен «умным» чатом, в котором можно описать проблему. Текст отправится на сервер, где произойдет синтаксический разбор и семантическое сопоставление с конкретной специальностью врача. В базе данных хранится информация обо всех посещениях и врачебных выписках (электронная медицинская карта), что позволяет, к примеру, напоминать пользователю о необходимости принять медикаменты.

Практическая ценность продукта заключается в *автоматизации* бизнес-процесса «прием пациентов», что приводит к *экономии* времени пациентов, обеспечению *высокой доступности* услуг и *оптимизации* трудовых затрат в медицинских учреждениях.

Технические детали

Клиентская часть выполнена нативными средствами на языке Swift 4.0 с использованием среды разработки Xcode. Основной архитектурой для всех модулей приложения взят VIPER (View-Interactor-Presenter-Entity-Router). Соблюдены принципы SOLID [1] и SOA, что позволяет в дальнейшем без трудностей расширять функциональные возможности приложения [2]. Дизайн выполнен с соблюдением Human Interface Guidelines от компании «Apple», что предоставляет пользователям привычный UX-дизайн (рис. 1).

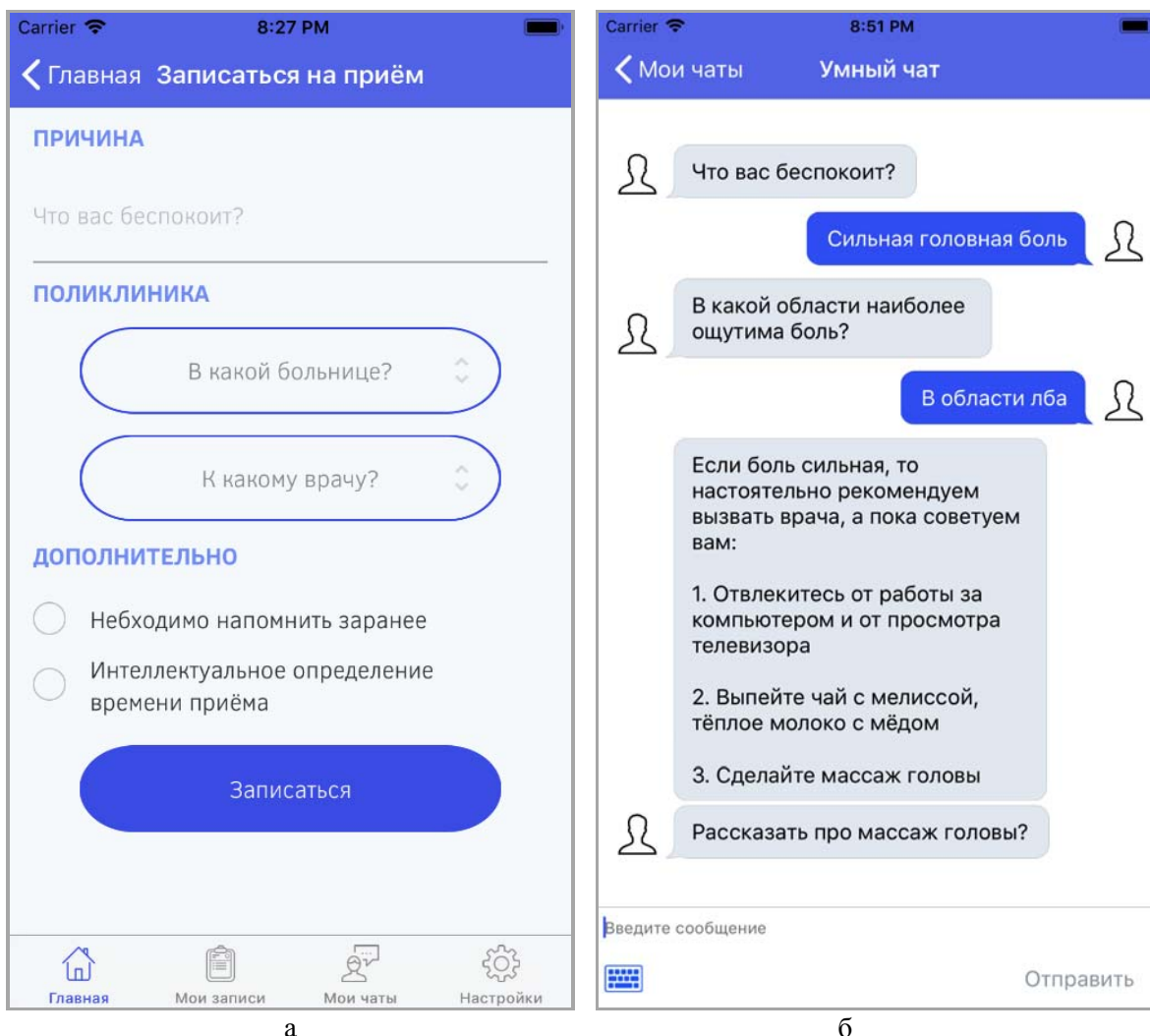


Рис. 1. Экран «Запись на прием» (а) и «Умный чат» (б)

Серверная часть реализована на языке Java 8 с использованием среды разработки IntelliJ IDEA. За архитектурную основу взят стиль REST (Representational state transfer) – в этой архитектуре данные передаются без дополнительных слоев, что делает ее менее ресурсоемкой в сравнении с SOAP или XML-RPC, здесь не нужно анализировать запрос, чтобы понять его природу и транслировать данные из одного формата в другой.

В качестве базы данных выбран PostgreSQL 10.

Математическая модель

Очевидно, что возможных решений (вариантов для приема) может быть несколько, и здесь мы имеем дело с комбинаторной задачей, которую для удобства можно записать в виде задачи удовлетворения ограничений (Constraint satisfaction problem, CSP) [3; 4]. В нашем случае CSP включает в себя:

- 1) variables – переменные;
- 2) domains – набор возможных значений;
- 3) constraints – список ограничений.

Правил всего два: для каждой переменной мы задаем набор возможных значений – переменным могут быть присвоены любые значения, а не только 1 или 0 («истина» или «ложь») [5]; а также у нас есть список ограничений, которым удовлетворяют исходные переменные.

Для нас решением задачи удовлетворения ограничений будет нахождение всех возможных значений, которые могут принимать переменные, с учетом существующих ограничений [6].

Термин «время для приема» следует разбить и рассматривать данную модель, как совокупность «дня» и конкретного «времени».

Выделим шесть переменных:

D – множество возможных значений времени, когда лечащий врач может принять пациента;

U – множество возможных дней для пациента;

C – множество общих значений времени, когда запись неосуществима (например, ночное время);

P – множество значений времени, когда пользователь не может посетить врача (множество строится на основе значений времени, указанных в заметках);

A – множество дополнительных значений времени, которое определяется на основании рабочего графика, предпочтительного времени и других подобных факторов, выбранных пользователем в мобильном приложении;

O – множество вычисляемых переменных-ограничений (например, если в заметках указана геолокация мест, то мы можем вычислить время, за которое пользователь доберется от текущего местоположения до цели, тем самым мы получим дополнительные ограничения).

Зададим ограничения, которые будем накладывать на переменные, чтобы получить актуальное множество решений:

$$\begin{aligned} D = U, \quad D \neq C, \quad U \neq C, \\ U \neq P, \quad U \neq A, \quad U \neq O. \end{aligned}$$

«Программирование в ограничениях» подразумевает собой использование декларативной формы программирования, что в отличие от императивного стиля позволяет нам в разы упростить задачу: нам нужно лишь описать проблему в общем случае, а всеми вычислениями и поиском решений будет заниматься так называемый решатель (Solver), который и содержит эффективные алгоритмы вычислений. К сожалению, исчерпывающих обзоров теории «удовлетворения ограничений» на русском языке нет, однако имеются публикации, которые освещают отдельные аспекты данной предметной области [8; 9].

После того как решатель вычислил множество значений времени, которые удовлетворяют нашим условиям – *мягким* и *глобальным* ограничениям [7], мы можем показать это множество пользователю, после чего осуществить запись на прием в медицинское учреждение.

Схема реализации

Рассмотрим одну из основных функций приложения – запись на прием (рис. 2). С помощью функции «запись на прием» пользователь попадает на экран, на котором расположены:

1) текстовое поле «что вас беспокоит?» – именно здесь пользователь может описать причину обращения, указать симптомы или другую информацию для врача;

2) поле выбора «в какой больнице?» – предоставляется список доступных медицинских учреждений, в которые интегрирован данный сервис;

3) поле выбора «к какому врачу?» – список доступных врачей для выбранной поликлиники;

4) поле «напомнить заранее» – при выбранном значении пользователь будет уведомлен о предстоящем визите заранее, посредством push-нотификации на мобильное устройство или почту;

5) поле «интеллектуальное определение времени приема» – при невыбранном значении пользователь может самостоятельно связаться с медицинским учреждением и согласовать время приема.

Если все-таки пациент выбрал «интеллектуальное определение времени», то на сервер отправляются задачи, которые были интегрированы с мобильного устройства и различных сервисов для планирования, а также отправляется множество выбранных пользователем «пред-

почтительных дат для приема». На «сервере» происходит сопоставление информации, полученной с «клиента», с графиком выбранного врача. Далее, на том же сервере, решатель (Solver) находит решение задачи программирования в ограничениях и отправляет их обратно «клиенту», где пользователь может выбрать наиболее подходящий вариант. После выбора информация вновь отправляется на «сервер», откуда попадает в медицинское учреждение, уведомляя тем самым врача о новой записи.

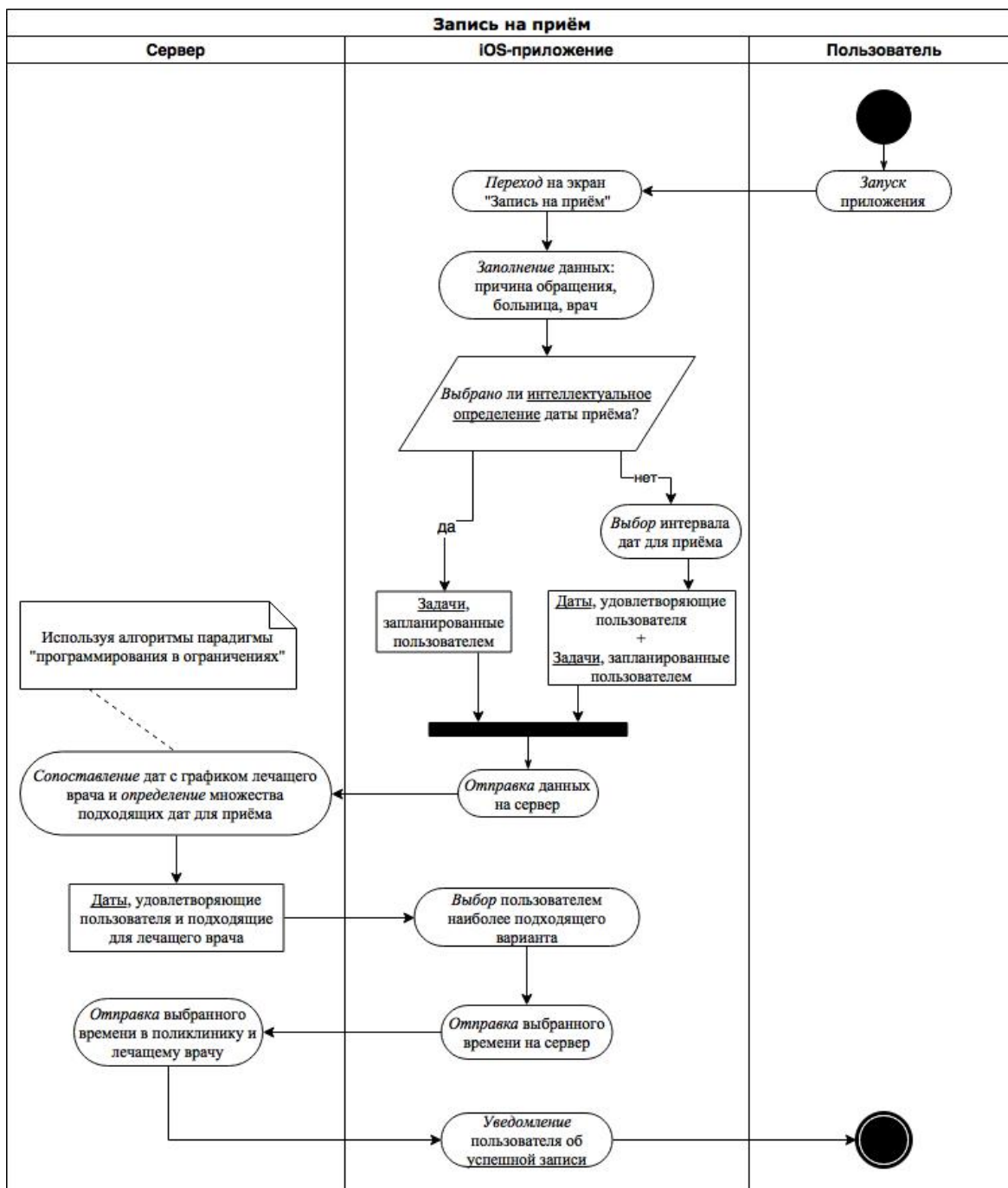


Рис. 2. Алгоритм истории «Запись на прием»

История с интеллектуальным чатом представляет собой творческую задачу. На данном этапе реализован простой синтаксический разбор текста, с последующим разбиением на лексемы, которые сопоставляются с семантическими командами (рис. 3).

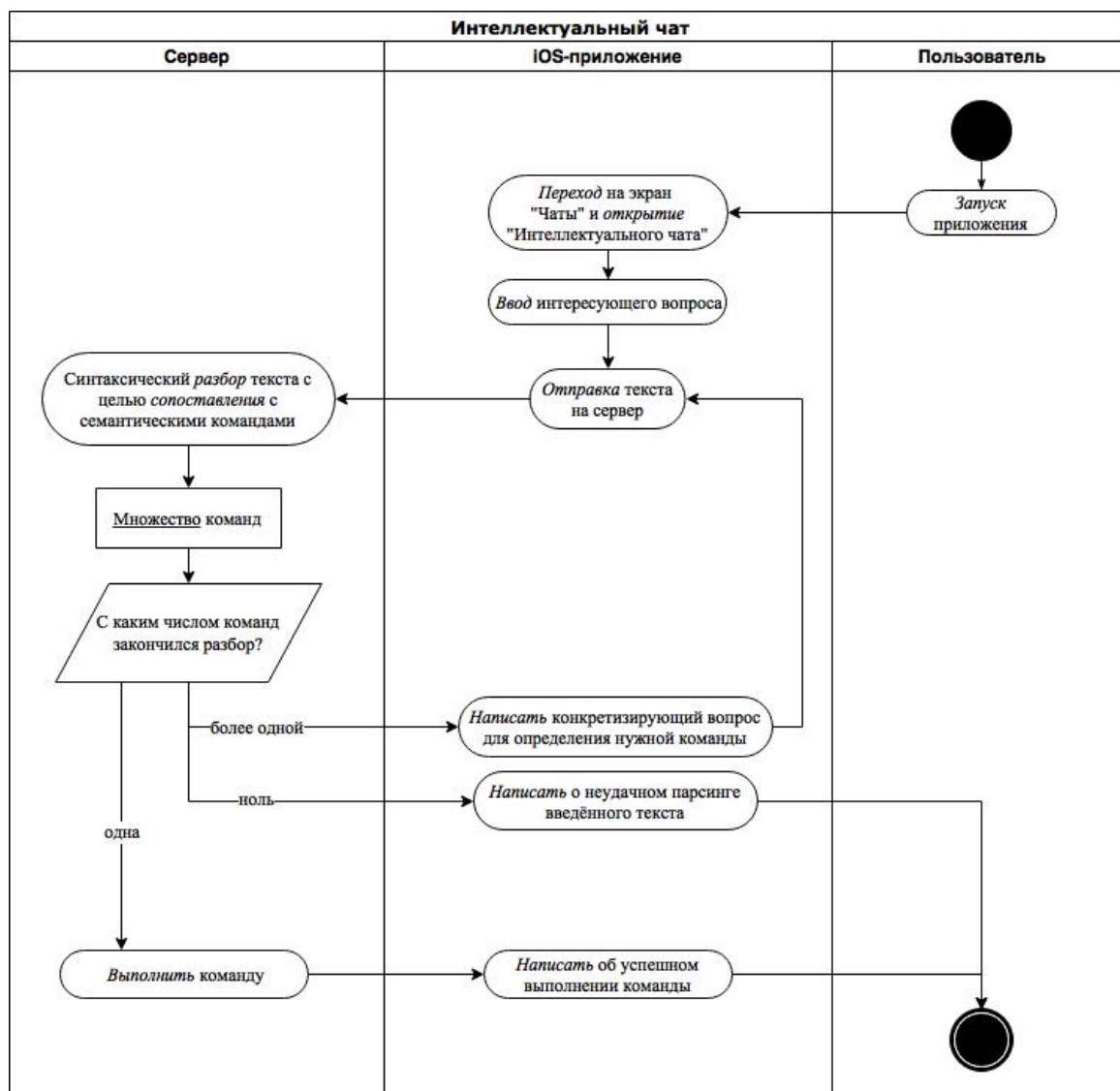


Рис. 3. Алгоритм работы интеллектуального чата

Заключение

Подводя итоги, можно сказать, что готовый продукт действительно является интеллектуальным помощником и промежуточным звеном между пациентами и медицинскими учреждениями. Теперь запись на прием становится простой и доступной для абсолютного большинства людей. Разработанный сервис особенно удобен для людей, у которых рабочий день не нормирован – им присущ плотный график с частыми форс-мажорными ситуациями, из-за чего данный сегмент зачастую пользуется сервисами-планировщиками, которые, в свою очередь, можно интегрировать с разработанным сервисом Medilux, что приведет к более точному определению наилучшего времени записи на прием.

Помимо прочего, интеллектуальный чат в приложении способствует сокращению обращений в поликлиники с целью быстрых вопросов или консультаций – теперь можно получить ответы (которые согласованы со специалистами) на большинство вопросов, не выходя из дома.

Электронная медицинская карта с полной историей записей избавляет от бумажной волокиты и позволяет напоминать пользователю о необходимости принять медицинские препараты через push-нотификации на мобильное устройство.

Сервис Medilux готов составить конкуренцию уже существующим сервисам в данной области. Разработанный продукт отличается уникальными особенностями, которые имеют векторы для развития, что приведет к более точным результатам определения времени записи на прием и расширению «словарного запаса» интеллектуального чата.

Список литературы

1. Макконел С. Совершенный код. Мастер-класс: Пер. с англ. М.: ИТД «Русская редакция»; СПб.: Питер, 2005. 896 с.: ил.
2. Холл Г. М. Адаптивный код: гибкое кодирование с помощью паттернов проектирования и принципов SOLID: Пер. с англ. 2-е изд. СПб.: Альфа-книга, 2017. 448 с.: ил.
3. Dechter R. Constraint processing. San Francisco: Morgan Kaufmann, 2003. 481 p.
4. Tsang E. Foundations of Constraint Satisfaction. New York: Academic Press, 1993. 421 p.
5. Dechter R., Frost D. Backtracking algorithms for constraint satisfaction problems. University of California, 1999. URL: <https://www.ics.uci.edu/~csp/r56-backtracking.pdf> (дата обращения 12.02.2018).
6. Beek P. van. Reasoning about qualitative temporal information // Artificial Intelligence. 1992. Vol. 58. P. 297–326.
7. Rossi F., Beek P. van, Walsh T. Chapter 4 Constraint Programming in Foundations of Artificial Intelligence // Handbook of Knowledge Representation. Eds. F. van Harmelen, V. Lifschitz, B. Porter. 2008. Vol. 3. P. 181–211.
8. Ушаков Д. М., Телерман В. В. Системы программирования в ограничениях (обзор) // Системная информатика: Сб. науч. тр. Новосибирск: Наука, 2000. Вып. 7: Проблемы теории и методологии создания параллельных и распределенных систем. С. 275–310.
9. Щербина О. А. Удовлетворение ограничений и программирование в ограничениях. URL: [http://www.intsys.msu.ru/magazine/archive/v15\(1-4\)/shcherbina-053-170.pdf](http://www.intsys.msu.ru/magazine/archive/v15(1-4)/shcherbina-053-170.pdf) (дата обращения 23.03.2018).

Материал поступил в редколлегию 24.03.2018

I. E. Bukshev

*Novosibirsk State University
1 Pirogov Str., Novosibirsk, 630090, Russian Federation*

*Center of Financial Technologies
2 Shaturskaya Str., Novosibirsk, 630055, Russian Federation*

bukshev@gmail.com

MEDILUX – SERVICE OF INTELLECTUAL FORMING THE SCHEDULE OF VISITING MEDICAL INSTITUTIONS

In this paper, a new type of service «intellectual registry» is offered. In the mobile application, there is a choice of employees of the medical institution. In addition to that, there is an analysis of the schedule of the selected doctor and free time of the patient using programming algorithms in limitations to determine the best time to book an appointment.

The knowledge base of the user's free time is formed based on tasks that are aggregated from the mobile device and popular task management services. In case the user does not know who to contact, the product is equipped with a «smart» chat where the problem can be described. To be exact, the text will be sent to the server where there will be a parsing and a semantic comparison with the specific qualification of the doctor.

The database stores information about all visits and medical statements (electronic medical records), which allows to remind the user about the need to take the medications.

The practical value of the product lies in the automation of the business process «reception of patients», which leads to saving patients' time, ensuring high availability of services and optimizing labor input in medical institutions.

Keywords: registry, intellectual, constraint, programming, Swift, iOS, Java.

References

1. McConnell S. Code Complete. 2nd ed. ISBN 0-7356-1967-0.
2. Hall G. M. Adaptive Code. 2nd ed. ISBN 978-1-5093-0258-1.
3. Dechter R. Constraint processing. San Francisco: Morgan Kaufmann, 2003, 481 p.
4. Tsang E. Foundations of Constraint Satisfaction. New York: Academic Press, 1993, 421 p.
5. Dechter R., Frost D. Backtracking algorithms for constraint satisfaction problems. University of California, Department of Information and Computer Science, 1999.
6. Beek P. van. Reasoning about qualitative temporal information. *Artificial Intelligence*, 1992, vol. 58, p. 297–326.
7. Rossi F., Beek P. van, Walsh T. Chapter 4 Constraint Programming in Foundations of Artificial Intelligence. *Handbook of Knowledge Representation*. Eds. F. van Harmelen, V. Lifschitz, B. Porter. 2008, vol. 3, p. 181–211.
8. Ushakov D. M., Telerman V. V. Sistemy programmirovaniya v ogranicheniyah (obzor) [Systems of programming in constraints (overview)]. Novosibirsk, Nauka, 2000, vol. 7, p. 275–310. (in Russ.)
9. Shcherbina O. A. Udovletvorenie ogranicheniy i programmirovaniye v ogranicheniyakh [Satisfaction of constraints and constraint programming]. URL: [http://www.intsys.msu.ru/magazine/archive/v15\(1-4\)/shcherbina-053-170.pdf](http://www.intsys.msu.ru/magazine/archive/v15(1-4)/shcherbina-053-170.pdf) (last access 23.03.2018). (in Russ.)

For citation:

Bukshev I. E. Medilux – Service of Intellectual Forming the Schedule of Visiting Medical Institutions. *Vestnik NSU. Series: Information Technologies*, 2018, vol. 16, no. 2, p. 41–48. (in Russ.)

DOI 10.25205/1818-7900-2018-16-2-41-48