

## АВТОМАТИЗИРОВАННАЯ ИНФОРМАЦИОННАЯ СИСТЕМА УЧЕТА И МОНИТОРИНГА УЧЕБНОГО ПРОЦЕССА В ВУЗЕ В УСЛОВИЯХ БАЛЛЬНО-РЕЙТИНГОВОЙ СИСТЕМЫ

А. И. Жданов\*, Н. А. Каплиева\*\*, И. В. Юргелас\*, Е. В. Корж\*,  
Р. Н. Фролов\*, Ю. Н. Юргелас\*, Д. А. Антонов\*\*

\*Воронежский государственный медицинский университет имени Н. Н. Бурденко

\*\*Воронежский государственный университет

Поступила в редакцию 18.10.2018 г.

**Аннотация.** В статье изложен опыт разработки информационной системы, включающей в себя серверную часть (Java 8) и клиентскую часть, реализованную в виде web-приложения (Dart) и мобильного клиента (Flutter). Настоящее приложение предназначено для автоматизации балльно-рейтинговой системы оценки знаний в высшем учебном заведении медицинского профиля. Сообщается о технологическом использовании встроенного сервера Undertow, MyBatis в виде реализации объектно-реляционного отображения (ORM), инструмента управления миграциями базы данных MyBatis Migrations, фреймворк Spring и AngularDart 4. Программа ориентирована на работу с СУБД Postgres с использованием драйвера JDBC. Описываются преимущества использования языка Dart по сравнению с JavaScript при разработке клиентской части приложения.

**Ключевые слова:** балльно-рейтинговая система, автоматизированные информационные системы в образовании, клиент-серверная архитектура, встроенный сервер Undertow, объектно-реляционное отображение, фреймворк Spring, фреймворк AngularDart 4, СУБД PostgreSQL, инструмент управления миграциями MyBatis Migrations, платформа Dart, платформа Flutter.

**Annotation.** The article describes the experience of development of the informational system containing back-end (Java 8) and front-end in the form of web-based app (Dart) and mobile app (Flutter). This application is created for the automation of five point and rating scheme of the evaluation process and records management in higher medical education. The article refers to the technological use of the inbuilt server Undertow, MyBatis in the form of the implementation of the object-relational mapping (ORM), control tool for the databases migration management – MyBatis Migrations, framework Spring and AngularDart 4. Software is aimed to support the database management system Postgres with the help of the JDBC driver. It describes the advantages of using the Dart language in comparison to the JavaScript in the development of the client part of the application.

**Keywords:** five point and rating system, automatic informational systems in educational process, client-server architecture, inbuilt server Undertow, object-relational mapping, framework Spring, framework AngularDart 4, PostgreSQL DBMS, MyBatis Migrations control tool, web-based app and mobile app (Dart and Flutter platforms).

### ВВЕДЕНИЕ

На протяжении последних 16 лет отечественная педагогика и педагогическая психология фронтально реализуют на всех формах обучения и основных образовательных про-

граммах систему балльно-рейтинговой оценки академической успешности и компетентностной сформированности. Этот циклический объективизированный процесс, носивший априори экспериментальный характер [1], призван был привнести положительные моменты для всех участников участников профессионального образования и обучения, обрести должную организационно-правовую прозрачность и адаптироваться к каждому

© Жданов А. И., Каплиева Н. А., Юргелас И. В., Корж Е. В., Фролов Р. Н., Юргелас Ю. Н., Антонов Д. А., 2018

структурному подразделению образовательного учреждения, реализующему научно-педагогическую деятельность [2].

Модульная система медицинского образования предполагает анализ и корректировку компетентностной сформированности обучающегося не только в ходе оценки уровня и разносторонности теоретических знаний, то есть клинического мышления, но и глубины освоения практических умений, что качественно отличает образование высшее медицинское, от классического университетского, и, в свою очередь, порождает необходимость создания и эргономического совершенствования многоуровневой балльно-рейтинговой системы. Последняя как система формирования кумулятивного показателя поэтапной циклической объективизированной оценки знаний обучающихся по преподаваемым на клинической кафедре дисциплинам предполагает следующие этапы формирования рейтинга: текущий рейтинг, промежуточный рейтинг, рейтинг по результатам промежуточной аттестации и окончательный рейтинг по дисциплине.

На каждом из этапов определения рейтинга неизменно выполняется условие равенства суммы весовых частей оценок единице.

Оценки, полученные при фронтальной проверке знаний, из пятибалльной системы переводятся в рейтинговые баллы и процент достижения цели.

Текущий рейтинг обучающегося определяется на контрольном занятии по результатам решения тестовых заданий (вес оценки 0,2), проверки уровня освоения практических умений (вес оценки 0,3) и собеседования в ходе решения профессиональных задач (вес оценки 0,5).

Промежуточный рейтинг по дисциплине складывается из промежуточного рейтинга за лекционный курс (вес оценки 0,05) и промежуточного рейтинга за практические занятия (вес оценки 0,95).

Промежуточная аттестация оценивается по методике, аналогичной оценке контрольного занятия, и определяется по результатам решения тестовых заданий (вес оценки 0,2), проверки уровня освоения практических умений

(вес оценки 0,3) и собеседования в ходе решения профессиональных задач (вес оценки 0,5).

Окончательный рейтинг по дисциплине определяется промежуточным рейтингом (вес оценки 0,6) и рейтингом по результатам промежуточной аттестации (вес оценки 0,4).

Рейтинговые баллы, полученные при расчете окончательного рейтинга по дисциплине, переводятся в пятибалльную систему для внесения декретированных записей в учетно-отчетную документацию – Зачетную книжку обучающегося, Зачетно-экзаменационную ведомость, Экзаменационный лист, Экзаменационный журнал.

Нивелировать не столь сложный с математической, сколь чрезвычайно кропотливый с технической точки зрения процесс расчета академического рейтинга при традиционно немалой трудоемкости преподаваемых в высшем медицинском учебном заведении дисциплин призваны различного рода программные продукты. Весьма удачным решением проблемы автоматизации данного процесса от обладания исходными данными – показателями успеваемости до получения конечного продукта – Ведомости учета текущего и промежуточного рейтинга, или электронного Журнала учета успеваемости и посещаемости, и Ведомости учета окончательного рейтинга представляется разработанный нами и совершенствовавшийся в ходе повседневной работы на базе кафедры госпитальной хирургии Воронежского государственного медицинского университета имени Н. Н. Бурденко в период с 2015 по 2018 гг. программный продукт «АТТЕСТАЦИЯ 2.15-2.17» [9, 10]. Перспективная бесперебойность работы балльно-рейтинговой системы кроется, по нашему мнению, в создании и совершенствовании узкоцелевых версий, модификаций и мобильных приложений данного продукта, которые позволяют формировать Аттестационный, или Ответный, лист как форму отчетности при процедуре промежуточной аттестации обучающегося по предмету и Отчет о текущей задолженности обучающегося для деканата по профилю специальности.

Учитывая вышеизложенное была поставлена **цель** автоматизации балльно-рейтин-

говой системы от момента ввода исходных показателей успеваемости и посещаемости до получения должной учетно-отчетной документации.

Для достижения поставленной цели нами были сформулированы следующие **задачи**:

1. Провести сравнительный анализ средств разработки клиентской части приложения (web-клиента и мобильного клиента).

2. Разработать приложение, способное выполнять нижеизложенные условия:

1) ввод, хранение, добавление, удаление, изменение информации об обучающихся и научно-педагогических работниках, вовлеченных в преподавание тех или иных дисциплин;

2) ведение учета посещаемости и успеваемости обучающихся;

3) расчет текущего, промежуточного и окончательного рейтинга по 5-ти и 100-балльной системам;

4) поиск информации по заданным критериям;

5) экспорт информации в формат (\*.xls) для формирования учетно-отчетной документации в развернутом и сокращенном вариантах при взаимодействии с табличным редактором (Excel).

## КРАТКИЙ ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Работа по созданию и внедрению информационных систем поддержки и мониторинга качества образования ведется во многих вузах России. На сегодняшний день рынок систем автоматизации работы высшего учебного заведения представлен широким спектром программных продуктов, например,

1) «Комкон: ВУЗ. Деканат 8» на платформе «1С:Предприятие 8»;

2) «БИТ: Учебная часть» на платформе «1С:Предприятие 8»;

3) информационная система для ВУЗов ЧДК: ВУЗ 8.1 фирмы ЗАО «Что делать Внедрение»;

4) система «Управление вузом» от компании Галактика.

Приведенные выше системы обладают широкой функциональностью. Однако можно отметить, что они оценивают наличие у обучающихся знаний по отдельным предметам, навыков и умений их применения в типовых предметных ситуациях, но нет определения рейтинга обучающихся по нескольким факторам.

Первые три решения не являются самостоятельным программным продуктом, так как для их работы требуется наличие установленной платформы «1С:Предприятие».

«Комкон: ВУЗ. Деканат 8» не имеет дополнительной защиты, рассматривается как самостоятельный модуль и не предполагает взаимодействия и обмена данными с другими конфигурациями.

Архитектура решения «ЧДК: ВУЗ 8.1» унифицирована с типовыми конфигурациями фирмы «1С», но не ориентирована на балльно-рейтинговую систему учета успеваемости.

Система Галактика «Управление вузов» предназначена для традиционных форм обучения, адресована образовательным учреждениям, которые обладают большим контингентом научно-педагогических работников и обучающихся и готовят специалистов в области как высшего, так и среднего профессионального образования. Система имеет сложный интерфейс, из-за чего предъявляется высокий уровень требований к компьютерной грамотности пользователей.

Подводя итоги, следует отметить, что рассмотренные системы не позволяют вести многокритериальную оценку (с весовыми коэффициентами) знаний обучающихся, не имеют расширения в виде мобильного клиента, что достаточно актуально в условиях рассредоточенности площадок проведения занятий в высшем медицинском учебном заведении, не предоставляют возможности построения динамически расширяющихся Excel-форматов ведомостей.

## РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Разработанная информационная система представляет собой клиент-серверное приложение.

Серверная часть приложения [9] реализована на платформе Java 8 [5]. Благодаря кросс-платформенности имеется возможность разрабатывать сервер на одной операционной системе, а непосредственно разворачивать приложение на другой. Для обеспечения необходимых функциональных возможностей используется встроенный сервер Undertow, MyBatis в виде реализации объектно-реляционного отображения (ORM), фреймворк Spring [4]. Приложение ориентировано на работу с СУБД PostgreSQL 9.6. Для взаимодействия с Postgre используется драйвер JDBC, для изменения структуры БД – инструмент управления миграциями MyBatis Migrations.

Реализация клиента [10] представлена в виде web-приложения и мобильного приложения.

На стороне web-клиента используется платформа Dart [3, 6]. Dart – молодой язык программирования с открытым исходным кодом, созданный Google. Код, написанный на Dart, может быть исполнен как на виртуальной машине Dart VM, так и может быть откомпилирован в JavaScript. Это дает возможность писать код на Dart для веб-клиентов, минуя JavaScript. Для разработки одностраничных веб-клиентов используется Dart-фреймворк AngularDart [7], обеспечивающий расширение браузерных приложений на основе MVC-шаблона, а также упрощающий тестирование и разработку.

Для создания мобильного клиента используется платформа Flutter [8], позволяющая разрабатывать кроссплатформенные мобильные приложения, обеспечивая при этом производительность на уровне нативных приложений. Код мобильного приложения пишется на Dart.

### Серверная часть

Для управления миграциями базы данных используется инструмент MyBatis Migration. Он позволяет легко поддерживать схему БД в актуальном состоянии, а также безопасно удалять внесенные изменения в случае нештатной ситуации.

Для увеличения производительности создается пул подключений к базе данных Hikari.

Вместо того, чтобы на каждый SQL запрос создавать соединение, в начале работы программы открывается указанное количество соединений к базе, и они держатся открытыми, до тех пор пока с ними что-нибудь не случится. В этом случае соединение закрывается и создается новое.

Приложение строится по многослойной архитектуре (рис. 1)

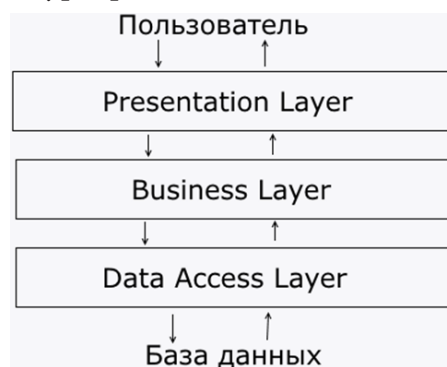


Рис. 1. Пример многоуровневой архитектуры

На нижнем уровне приложения происходит взаимодействие с базой данных посредством SQL-запросов, а также отображение объектов из базы в объекты объектно-ориентированной модели.

После выполнения SQL-запроса в реализованном методе идет проверка, какой тип данных он возвращает и делается попытка отобразить полученный результат из базы в объекты этого класса (название колонки отображается в название поля класса). Если SQL-запрос содержит параметры, то они берутся из параметров метода.

Уровнем выше располагается *Business Layer* (слой бизнес логики). Здесь выполняются манипуляции с данными и реализуется большинство требований к приложению. Данный слой обладает информацией о нижестоящем слое, поэтому может его использовать. Классы, реализующие бизнес логику, называются сервисными классами (в названии класса обычно фигурирует слово Service) и в простейшем случае реализуют CRUD (create, read, update, delete) операции, то есть повторяют соответствующий интерфейс из слоя доступа к данным. Дополнительной функциональностью, которой может обладать сервис, является кеширование данных.

Слой представления данных оперирует сущностями, которые тесно связаны с транспортным уровнем, и используется для получения данных от клиента и передачи их слою бизнес-логики.

Слой представления данных реализован с помощью фреймворка Spring MVC, обеспечивающего архитектуру паттерна Model-View-Controller (Модель-Отображение-Контроллер) при помощи слабо связанных готовых компонентов.

Вся логика работы Spring MVC построена на основе DispatcherServlet, который принимает и обрабатывает все HTTP-запросы (из UI) и ответы на них.

Для обеспечения безопасности приложения, разграничения прав доступа, авторизации и т.д. используется фреймворк Spring Security. Spring Security в общем случае представляет собой систему фильтров HTTP-запросов. Фильтры организуются в цепочки, которые либо разрешают запросу пройти дальше по цепочке, либо отклоняют запрос и

возвращают код ошибки. Запрос, прошедший все фильтры, отправляется на последующую обработку Spring MVC.

Одной из задач приложения является генерация отчетности в формате \*.xlsx. Пример ведомости учета текущего и промежуточного рейтинга представлен на рис. 2. Задача выгрузки данных в отчет решается с помощью библиотеки Apache POI, которая предоставляет удобный интерфейс для чтения, модификации и сохранения файлов электронных таблиц.

При поступлении запроса на формирование ведомости, создается временный файл, который является копией нужного шаблона и удаляется из системы спустя некоторое время. С помощью библиотеки Apache POI данные вставляются в таблицу, после чего заполненная таблица сохраняется во временном файле. Из соображений безопасности ответом сервера на запрос генерации отчета является путь к файлу, используя который можно получить необходимый отчет.

ФГБОУ ВО ВГМУ им. Н.Н. Бурденко Минздрава России												
Кафедра госпитальной хирургии												
ВЕДОМОСТЬ												
учета текущего и промежуточного рейтинга												
Учебный год	2016/2017	Семестр	4	Дата заполнения			12.01.2017 г.					
Факультет (институт, специальность)	МИМОС, Лечебное дело			Курс	4	Группа	401					
Дисциплина	Хирургические болезни			Цикл	1							
№№ пп	Фамилия и инициалы студента	Лекционный курс, баллы	Контрольные занятия, №№ / баллы									Промежуточный рейтинг по дисциплине, баллы
			1			2			3			
			ТК	ПУ	ПЗ	ТК	ПУ	ПЗ	ТК	ПУ	ПЗ	
1.	Константиновский К.К.											
Примечание:		ТК – тестовый контроль, ПУ – практические умения, ПЗ – профессиональная задача										
Преподаватель кафедры госпитальной хирургии, к.м.н., профессор		Александровский А. А.										

Рис. 2 Пример сформированной Ведомости учета текущего и промежуточного рейтинга виртуальной группы обучающихся (полная версия)

### **Клиентская web-часть**

Клиентская web-часть написана на языке Dart с использованием фреймворка AngularDart, в который входит инструмент для управления зависимостями между компонентами внутри программы (Dependency Injection).

Одна из особенностей Angular заключается в том, что представление автоматически реагирует на смену модели. Например, если к списку факультетов на программном уровне добавить еще один, то он автоматически отобразится в таблице.

Приложение разделено на три слоя: слой представления данных, слой обработки данных и слой сущностей.

Слой сущностей определяет структуру сущностей, которыми оперирует приложение (обучающийся, преподаватель и т. д.).

Слой обработки данных представляет собой группу сервисов для каждой сущности, в обязанности которой входит конвертация данных из json-формата в обычный Dart-объект и наоборот (для этого используется библиотека Dartson), загрузка данных с сервера, управление их кэшированием (если есть необходимость в кэше) и сохранение данных на сервер.

Слой представления данных реализует паттерн MVC, отвечающий за логику работы и изменение представления в соответствии с текущим состоянием модели.

В приложении предусматривается отображение сообщений в виде всплывающей информации о возникшей ошибке или об успешном завершении операции (например, сохранения) и т. д. Так как у AngularDart нет подобных компонентов, поэтому был разработан компонент уведомлений.

Основной сущностью компонента уведомлений является само уведомление, состоящее из 3 полей: тип уведомления (ошибка, успех, информативное, предупреждение), заголовок и сообщение.

У этого компонента есть два настраиваемых параметра: максимальная длина списка уведомлений и количество времени в секундах, в течение которого уведомление будет показываться пользователю.

Для входа в приложение (ввод логина и пароля) было решено сделать отдельную легковесную страницу, созданную при помощи стандартных средств: HTML, CSS, JS + JQuery. Кроме очевидных преимуществ этого решения в виде более быстрой загрузки, уменьшения количества трафика и ряда других, есть еще одно, связанное с безопасностью. Пользователь, у которого нет доступа к рейтинговой системе, не сможет получить ни ее код, ни какую-либо информацию о ней.

### **Мобильный клиент**

Мобильное приложение создано с использованием нестандартных технологий. Вместо нативных Java и Swift для Android и iOS соответственно, использовалась технология Flutter от компании Google.

Особенностью Flutter является то, что, написав код приложения один раз, его можно скомпилировать сразу для двух устройств. При этом скорость работы приложений, сделанных при помощи Flutter, остается на уровне нативных реализаций.

Для разработки приложений на основе платформы Flutter используется также язык Dart, что означает возможность использования многих наработок, за исключением слоя представления данных, из веб-версии приложения.

Одной из особенностей, которая ускоряет разработку, является возможность перекомпилировать приложение «на лету». При разработке проекта и запуске его на эмуляторе, можно в коде поменять фрагмент и увидеть результат этой замены на экране эмулятора без перезапуска эмулятора или приложения.

И, наконец, не менее заметной особенностью является то, что данная платформа «из коробки» предоставляет основные компоненты (поле ввода, флажки и т.д.) в рекомендуемом стиле целевой платформы (Material Design для Android и Cupertino для iOS).

Мобильное приложение построено по той же трехслойной схеме, что и веб-версия: слой представления данных, слой обработки данных и слой сущностей.

Слой сущностей определяет структуру сущностей, которыми оперирует приложение

(обучающийся, преподаватель и т. д.). Однако в данном случае Flutter не поддерживает библиотеку Dartson для сериализации объектов в json, так как процесс компиляции проекта в этом случае сильно отличается от того, который происходит при компиляции веб-версии. Именно поэтому для сериализации используется инструмент, который является частью библиотеки convert языка Dart. Данная библиотека работает не с объектами, а со структурой данных под названием Map, следовательно, необходимо дополнительно делать конвертацию между объектом и Map, и наоборот.

Слой обработки данных представляет собой группу сервисов для каждой сущности, в обязанности которой входит: загрузка данных с сервера, управление их кэшированием (если есть необходимость в кэше) и сохранение данных на сервер. Но здесь, по сравнению с веб-версией, отсутствует dependency injection контейнер, что предопределило разработку упрощенной версии статических полей класса DIContainer. В остальном реализация ничем не отличается от веб-версии.

Слой представления данных для мобильного приложения сильно отличается от аналогичного слоя для веб-приложения как по структуре, так и по подходу. Flutter обеспечивает более декларативный подход к разработке пользовательского интерфейса. Если в версии приложения на Angular единицей пользовательского интерфейса был компонент, то для Flutter он называется виджетом. Последние бывают двух видов: stateless и statefull. Stateless виджеты не сохраняют состояние, т. е. не имеют возможности перерисовки при смене внутреннего состояния. Если stateless виджет является списком обучающихся, то при добавлении нового обучающегося обновления списка не происходит. В случае со statefull виджетами все наоборот, они хранят состояние и могут перерисовываться при смене внутреннего состояния.

## **ЗАКЛЮЧЕНИЕ**

Близость к разрешению проблемы создания нормативно-правовой базы балльно-рейтинговой системы оценки академической успешности и компетентностной сформированности обучающегося как на ведомственном уровне, так и в преломлении к индивидуально-вузовской адаптации позволяет перераспределить все силы и внимание на сосуществующую проблему автоматизации, делопроизводства и документооборота этой же системы. Достижения ключевого звена обозначенной цели следует добиваться, прежде всего, через решение условия эргономичности – создание и совершенствование различного рода программных продуктов, их модификаций, узкоцелевых версий и мобильных приложений, примером чего может служить представленная информационная система АТТЕСТАЦИЯ 2.17.

Сравнительный анализ в целях выбора наиболее удобного языка реализации клиентской части приложения и последующего переноса решения на мобильную платформу выявил неоспоримые преимущества языка Dart над JavaScript.

Язык Dart находится в стадии активного развития. Для систем, которые планируют развиваться и над которыми будут работать несколько команд разработчиков, преимущества языка Dart над JavaScript (в веб-клиентской части) не вызывают сомнений.

JavaScript является динамически типизированным языком (при объявлении переменной тип переменной не указывается). В связи с этим в JavaScript недостаточно информации для самодокументации, и для больших проектов требуется ведение дополнительной документации.

Проверка типа осуществляется на этапе непосредственного выполнения программы, что влечет за собой более позднее обнаружение ошибок несовместимости типов, а это в свою очередь может привести к усложнению решения и увеличению времени разработки программного продукта.

В языке JavaScript имеется возможность динамически изменять структуру объектов

(добавлять или удалять поля, изменять реализацию методов и т. д.). Если такое изменение произойдет по ошибке, то такую ошибку будет очень трудно обнаружить, особенно в больших проектах с большой кодовой базой.

Язык Dart лишен этих недостатков, так как он статически типизирован.

Тип переменной указывается при ее объявлении (есть тип `dynamic`, который аналогичен `var` в JavaScript, но, начиная с версии языка Dart 2.0, тип указывается обязательно). Таким образом, новичку будет легче войти в проект, не изучая при этом документацию.

В Dart нельзя динамически модифицировать объект, следовательно, это исключает ошибки такого рода.

Разработчик, знающий популярные ООП языки (C++, Java, C#), без особых проблем освоит Dart, чего нельзя сказать про JavaScript. Это добавляет некоторую гибкость в разработку системы.

С точки зрения мобильной разработки корректнее рассматривать связку языка Dart и технологии Flutter, а не отдельный язык.

Явным преимуществом является то, что приложение можно написать один раз, а скомпилировать на две основные системы (Android и iOS), обеспечивая кроссплатформенность приложения.

Идея кроссплатформенности в мобильной разработке не нова. Более известным решением является React Native, но программирование на этой платформе осуществляется на языке JavaScript. При этом React Native не обеспечивает скорость работы приложения сравнимую с нативной реализацией, в отличие от Flutter.

Использование Dart как для реализации web-клиента, так и для мобильной части, дает возможность повторного использования кода, что значительно сокращает время на разработку и отладку приложения.

## СПИСОК ЛИТЕРАТУРЫ

1. О проведении эксперимента по введению рейтинговой системы оценки успеваемости студентов вузов // Приказ МО РФ от 11 июля 2002 года №2654 / Бюллетень Министерства просвещения. – № 11. – 2002. – С. 13–16.

2. Система балльно-рейтинговой оценки академической успешности и компетентностной сформированности обучающегося в высшем учебном заведении / А. И. Жданов, И. В. Юргелас, Р. Н. Фролов, Ю. Н. Юргелас // Современные наукоемкие технологии. – 2017. – № 6. – С. 122–127.

3. *Баккет К.* Dart в действии. / К. Баккет. – М. : ДМК Пресс, 2013. – 528 с.

4. *Шеффер К.* Spring 4 для профессионалов. / К. Шеффер. – 4-е издание. – М. : Издательский дом «Вильямс», 2017. – 752 с.

5. *Блох Д.* Java. Эффективное программирование. / Д. Блох. – 2-е издание – М. : Издательство «Лори», 2014 – 310 с.

6. Dart for the web. – URL: <https://webdev.dartlang.org/guides/get-started>

7. About AngularDart Components. – URL: <https://webdev.dartlang.org/components>

8. Flutter. – URL: <https://flutter.io>

9. АТТЕСТАЦИЯ 2.17.1 : свидетельство о государственной регистрации программы для ЭВМ № 2018611434 / А. И. Жданов, И. В. Юргелас, Р. Н. Фролов, Ю. Н. Юргелас, Е. В. Корж, Н. А. Каплиева, Д. А. Антонов; заявитель и правообладатель ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России). – Москва, 2018. – (заявка № 2017662622, заявл. 05.12.2017, зарег. 01.02.2018).

10. АТТЕСТАЦИЯ 2.17.2 : свидетельство о государственной регистрации программы для ЭВМ № 2018611435 / А. И. Жданов, И. В. Юргелас, Р. Н. Фролов, Ю. Н. Юргелас, Е. В. Корж, Н. А. Каплиева, Д. А. Антонов; заявитель и правообладатель ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России). – Москва, 2018. – (заявка № 2017662620, заявл. 05.12.2017, зарег. 01.02.2018).



**Жданов Александр Иванович** – д-р мед. наук, профессор, ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России, декан лечебного факультета, заведующий кафедрой госпитальной хирургии.

Тел.: +7(473)257-97-17

E-mail: lech@vsmaburdenko.ru

**Каплиева Наталья Алексеевна** – канд. физ.-мат. наук, доцент кафедры математического обеспечения ЭВМ, факультет прикладной математики, информатики и механики, Воронежский государственный университет.

Тел.: 8(473)220-82-98

E-mail: kaplieva@amm.vsu.ru

**Юргелас Игорь Владимирович** – канд. мед. наук, ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России, доцент кафедры госпитальной хирургии.

Тел.: +7(473)257-97-17

E-mail: iyurgelas@yandex.ru

**Корж Елена Викторовна** – канд. мед. наук, ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России, доцент кафедры акушерства и гинекологии №1.

Тел.: +7(473)257-96-74

E-mail: elenakorzh2012@mail.ru

**Фролов Роман Николаевич** – канд. мед. наук, ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России, доцент кафедры госпитальной хирургии.

Тел.: +7(473)257-97-17

E-mail: vsurgeon@yandex.ru

**Юргелас Юлия Николаевна** – канд. мед. наук, ФГБОУ ВО ВГМУ им. Н. Н. Бурденко Минздрава России, ассистент кафедры госпитальной терапии и эндокринологии.

Тел. +7(473)257-96-98

E-mail: jyurgelas@yandex.ru

**Антонов Денис Андреевич** – магистр по направлению Фундаментальная информатика и информационные технологии, инженер-программист, компания «Рексофт».

Тел.: +7 (951) 548-00-65

E-mail: den007230@gmail.com

**Zhdanov A. I.** – Doctor of Medicine, Professor, Voronezh N. N. Burdenko State Medical University, Dean of the General Medicine Faculty, Head of the Hospital Surgery Chair.

Tel.: +7(473)257-97-17

E-mail: lech@vsmaburdenko.ru

**Kaplieva N. A.** – Candidate of Physics-math. Sciences, Associate professor the department of Mathematical Support of electronic computing machines, the Faculty of applied mathematics, information science, and mechanics, the Voronezh State University.

Tel.: +7(473)220-82-98

E-mail: kaplieva@amm.vsu.ru

**Yurgelas I. V.** – Candidate of Medicine, Voronezh N. N. Burdenko State Medical University, Associate Professor of the Hospital Surgery Chair

Tel.: +7(473)2579717

E-mail: iyurgelas@yandex.ru

**Korzh E. V.** – Candidate of Medicine, Voronezh N. N. Burdenko State Medical University, Associate Professor of the Gynecology and Obstetrics #1 Chair

Tel. +7(473)2579674

E-mail: elenakorzh2012@mail.ru

**Frolov R. N.** – Candidate of Medicine, Voronezh N. N. Burdenko State Medical University, Associate Professor of the Hospital of the Hospital Surgery Chair

Тел.: +7 (473) 257-97-17

E-mail: vsurgeon@yandex.ru

**Yurgelas J. N.** – Candidate of Medicine, Voronezh N.N. Burdenko State Medical University, Assistant of the Hospital Therapy and Endocrinology Chair

Tel. +7 (473) 257-96-98

E-mail: jyurgelas@yandex.ru

**Antonov D. A.** – Master of Fundamental Informatics and information technologies, software engineer, company Reksoft.

Tel.: +7 (951) 548-00-65

E-mail: den007230@gmail.com