

УДК 004.75:004.455

**С. К. Грудо, А. А. Молдованов**

Белорусский государственный технологический университет

**РАЗРАБОТКА МОБИЛЬНОЙ ВЕРСИИ ИНТЕРНЕТ-РЕСУРСА  
«500-ЛЕТИЕ КНИГОПЕЧАТАНИЯ В БЕЛАРУСИ»  
НА БАЗЕ ОПЕРАЦИОННОЙ СИСТЕМЫ ANDROID**

В статье решается задача разработки мобильного приложения на платформе Android для информационного ресурса Национальной библиотеки Беларуси, посвященного 500-летию белорусского книгопечатания. Выбор платформы обусловлен массовостью и широким распространением среди пользователей современных устройств на базе ОС Android, которая характеризуется высокой доступностью средств разработки и поддержанием различных многофункциональных приложений. Мобильное приложение адаптировано для различных версий мобильных устройств на данной платформе и предназначено для отображения цифровых материалов из базы данных серверной части информационной системы.

В качестве интегрированной среды разработки для клиентской стороны использовалась IDE Android Studio. Клиентская часть информационной системы реализована на объектно-ориентированном языке программирования Java. Совместное использование программного кода языка Java и XML-верстки позволило создать гибкий и расширяемый интерфейс для различных разрешений экрана мобильных устройств.

В основе разработки приложения положены алгоритмы, обеспечивающие широкий спектр функциональных возможностей: отображать данные различных типов, осуществлять их кеширование, выполнять постраничную навигацию с нижней панелью предварительного просмотра, поддерживать работу с географическими данными. Результаты проведенных тестов по использованию приложением ресурсов мобильной системы свидетельствуют о стабильности его работы, невысоком потреблении энергоресурсов и ресурсов памяти, экономии интернет-трафика.

**Ключевые слова:** мобильное приложение, интегрированная среда разработки, интерфейс, язык программирования, сетевой ресурс, мобильное устройство.

**S. K. Grudo, A. A. Moldovanov**

Belarusian State Technological University

**DEVELOPMENT OF MOBILE VERSION OF THE INTERNET-RESOURCE  
“THE 500TH ANNIVERSARY OF PRINTING IN BELARUS”  
BASED ON ANDROID OPERATING SYSTEM**

There is representing the solving of the problem of developing a mobile application that acts with internal information database of the National Library of Belarus dedicated to the 500th anniversary of Belarusian printing. The platform choosing is determined by its popularity and wide distribution among users of modern devices based on the Android OS, which is characterized by high availability of development tools and maintenance of various multifunctional applications. The mobile application is adapted for different versions of mobile devices on this platform and is intended for displaying a data from the server database.

Android Studio was used as an integrated development environment for the client side. The client part of the information system is implemented using the object-oriented programming language like Java. Java programming language and XML-layout allowed to create a flexible and extensible interface for various resolutions of the screen of mobile devices.

The development of applications is based on algorithms that provide a wide range of functionality: display different types of data, make it to be caching, perform page navigation with the bottom viewer panel, support the working with geodata. The results of the tests of application resources using show its stability of its operation, low energy and memory resources consumption that allow to save Internet traffic.

**Key words:** mobile application, integrated development environment, interface, programming language, network resource, mobile device.

**Введение.** В настоящее время главной тенденцией при работе с информацией является преобразование ее в цифровую форму с последующим хранением в электронном виде. Полу-

чение огромных объемов цифровых данных осуществляется из различных баз данных (БД) через программы, выполняющие роль клиентской стороны в распределенных информацион-

ных системах [1]. На сегодняшний день перспективным инструментом для комфортного и оперативного распространения информации являются мобильные приложения, представляющие собой специализированное программное обеспечение для работы на смартфонах, планшетах и других мобильных устройствах [2]. Многие мобильные приложения предустанавливаются на самом устройстве или загружаются на него из онлайн-официальных магазинов приложений, таких как App Store, BlackBerry App World, Google Play, Windows Phone Store и др.

Из существующих на современном рынке мобильных операционных систем именно платформа Android имеет широкую популярность пользователей в силу своей открытости, наличия относительно недорогих девайсов и современных средств разработки приложений. Android-система основана на ядре Linux и собственной реализации виртуальной машины Java от Google [3], что позволяет ей создавать Java-приложения, которые управляют устройством через разработанные Google-библиотеки. Таких библиотек существует большое множество, например Bionic, Surface Manager, FreeType, SSL и др. Кроме того, ОС Android доступна для различных аппаратных платформ, таких как ARM, MIPS, x86, что позволяет ей сохранять свою популярность и в среде разработчиков.

Для Республики Беларусь 500-летие белорусского книгопечатания – это важное событие, охватившее информационное пространство на долгое время. С целью популяризации истории отечественного книгопечатания и культуры, а также лаконичного донесения информации в данной предметной области для людей всех возрастных категорий авторами данной статьи предложена разработка информационной системы с обширной базой данных и удобным клиентским приложением на платформе Android. В задачи программного продукта входит освещение не только текущих мероприятий, связанных с юбилейной датой, но и текущее информирование, обеспечение электронного доступа к информационным ресурсам Национальной библиотеки Беларуси, поддержание коммуникационных связей среди заинтересованных лиц.

**Основная часть.** Реализуемая информационная система представляет собой комплекс [4], состоящий из сервера с БД, системы управления БД (СУБД), веб-сервера, единого протокола обмена информацией клиент – сервер, а также самого мобильного приложения. В общем, разработка такой системы включает последовательное выполнение следующих технических решений:

– согласование типа, количества и предметной области информации, которая будет храниться в БД. Основной тип цифровых данных –

это изображения различных цифровых форматов, текстовая информация в различных способах ее представления и хранения;

– построение основных моделей данных, по результатам которого выбран реляционный тип БД и СУБД MySQL;

– выбор веб-сервера (программной части) исходя из требуемой гибкости и скорости работы системы, а конкретнее: взаимодействия между клиентом и веб-сервером. Понятие гибкости в данном случае отражает предоставляемые программисту возможности в написании и модификации исходного кода и то, как скорость работы веб-сервера зависит от выбранного фреймворка. С учетом специфики предметной области приложения структурирование веб-части было выполнено при помощи фреймворка Laravel, дающего возможность расширения функционала (интеграции дополнительных модулей) без существенных затрат на изменение текущей системы, а также риска возникновения нежелательных потерь для веб-ресурса [5];

– выбор типа клиентских устройств и платформы под них. Классическими клиентскими устройствами являются десктоп-ПЭВМ и мобильные ЭВМ. В случае первых клиентская часть реализуется посредством браузера, в случае вторых – чаще всего посредством мобильного программного обеспечения в виде отдельного приложения.

Разработка мобильного приложения начинается с определения его программной структуры согласно поставленным задачам. В данном случае мобильное программное обеспечение должно поддерживать загрузку разнотипных данных с веб-сервера, их корректное кэширование, а также отображение. При этом, выполняя свои функции, оно должно потреблять минимальное количество аппаратных ресурсов устройства, за исключением ситуаций с ошибками, которые должны быть обработаны, чтобы не допустить некорректного закрытия программы во время ее выполнения.

Для согласования протокола обмена информацией между сервером и клиентом было принято классическое решение – протокол HTTP. По данному протоколу осуществляется отправка запросов и ответов на них. Сами же данные могут передаваться в нескольких видах, например в виде XML- либо JSON-формате.

После этого начинается непосредственно разработка всех составляющих элементов информационной системы (рис. 1). Для взаимодействия сервера с клиентской частью реализовывался Application Programming Interface (API). Некоторые из элементов системы разрабатывались параллельно, что значительно ускорило процесс реализации поставленных задач.

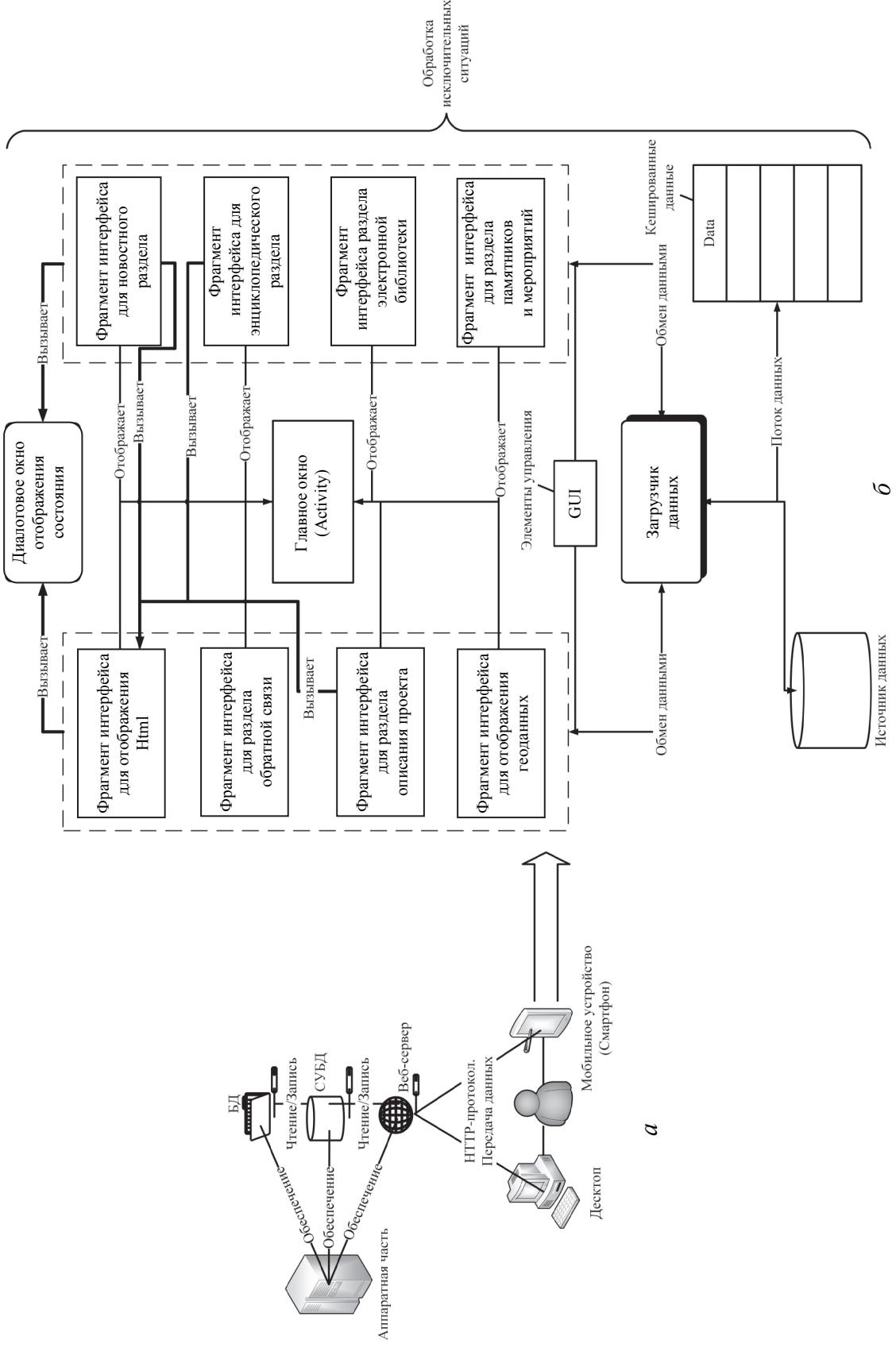


Рис. 1. Общая структура Android-приложения:

a – общая структура информационной системы; б – структура мобильного приложения

В качестве основного языка программирования для написания клиентской части информационной системы использовался объектно-ориентированный язык программирования Java [1] в силу того, что именно он является основным языком разработки под платформу Android. Основным алгоритм работы программы представляет собой комплекс функциональных блоков, каждый из которых выполняет свое назначение. Для этого создавались программные пакеты тематической направленности решаемых задач (рис. 2). Причем для исключения часто встречающихся недочетов аналогичных по структуре систем авторами статьи разработаны кастомизированные алгоритмы кэширования и восстановления информации, позволяющие ускорить работу приложения, а также оптимизировать затраты ресурсов устройств.

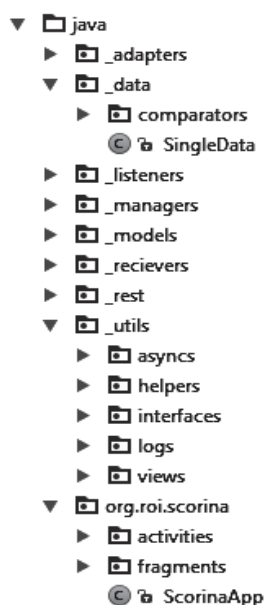


Рис. 2. Архитектура функциональных пакетов

Каждый разработанный пакет выполняет строго возложенные на него обязанности, начиная от работы с сервером и обработки данных вплоть до роли слушателя сообщений от системы:

- `_adapters` хранит в себе классы для адаптации моделей к их визуальной составляющей;
- `_data` содержит классы, необходимые для сортировки данных, поступающих от сервера и для их кэширования в памяти с целью экономии пользовательского мобильного интернет-трафика;
- `_listeners` содержит исходный код обработки изменения масштаба View-компонентов (представлений) при касании пользователем экрана мобильного устройства;
- `_managers` хранит исходный код классов менеджеров работы с объектами карты;

- `_models` хранит модели описания данных, используемых в приложении. Многие из данных классов составляются по аналогии с таблицами из реализованной на стороне сервера базы данных. Это необходимо для гарантии соответствия полей при парсинге json-файла, получаемого от сервера;

- `_rest` осуществляет преобразование json-формата в объекты классов из пакета `_models` посредством рефлексии в Java и свободно расширяемой библиотеки Retrofit2. Кроме того, данный пакет содержит классы для работы с сервером. В нем прописаны наименования полей таблиц, базовый URL-адрес, способ получения информации, методы запроса данных и т. п. Именно классы из данного пакета гарантируют корректную работу приложения с сервером, обмен между ними данными и обработку исключительных ситуаций, таких как недоступность сервера, превышение интервала ожидания, неверный запрос и подобное;

- `_receivers` предназначен для хранения классов, отвечающих за прослушивание сообщений от операционной системы. Последняя же позволяет подписываться на них. Каждое из сообщений имеет фильтр по событию;

- `_utils` отвечает за основные View-компоненты, которые чаще всего видит пользователь приложения.

Каждый созданный пакет имеет архитектуру классов, а каждый класс имеет свой алгоритм работы в совместном функционировании с другими классами, например:

- алгоритм метода разбиения строки с разделителем на массив подстрок;
- алгоритм обработки касаний пользователем дисплея в режиме просмотра изображений либо представлений, поддерживаемых изменение масштаба;
- алгоритм обработки двойных кратковременных касаний пользователем дисплея устройства.

Следует отметить, что некоторые из классов имели большое количество общих логических решений, а другие могли наследоваться вместе с их логикой.

В данной работе для реализации интерфейса было предложено решение совместного использования XML-верстки и программного кода на Java. Связывание XML-представления интерфейса и исходного кода на Java осуществлялось в классе `MainACTivity.java`. Для этого в методе `onCreate()` необходимо вызывать метод `setContentview()`, в который в качестве параметров передается уникальный ID-номер XML-файла. У любого объекта View может быть связан с ним целочисленный идентификатор, который служит для обозначения уни-

кальности объекта View в иерархии. Во время компиляции приложения этот идентификатор используется как целое число. Однако идентификатор обычно назначается в файле XML-макета в виде строки в атрибуте ID. Этот атрибут XML является общим для всех объектов View (определенных классом View), который используется довольно часто. Данный номер генерируется самой средой разработки, а разработчик может получить его, обратившись к системе через его имя.

Приложение, посвященное 500-летию белорусского книгопечатания, включает несколько тематических разделов, составленных совместно с администрацией Национальной библиотеки Беларуси.

Для отображения метаинформации и самого контента каждой новости создан файл XML-разметки, который отвечает за отдельно взятое представление в списке. В качестве списка использован компонент RecyclerView из стандартной библиотеки Android [3].

Представление на новость в приложении содержит текстовые поля TextView, изображение предварительного просмотра ImageView, а также элемент отображения состояния загрузки ProgressBar (рис. 3). Компонент RecyclerView позволяет задавать количество колонок для отображаемого списка.

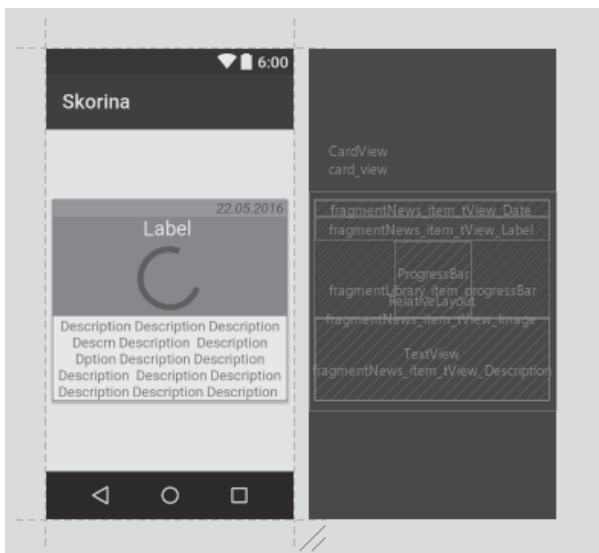


Рис. 3. Шаблон визуального компонента для каждой новости

Применяемая интегрированная среда разработки (ИСП) Android Studio позволяет использовать концепцию WYSIWYG для разработки пользовательского интерфейса [6]. Это дает возможность еще до компиляции проекта наблюдать будущий визуальный компонент из самой среды.

Каждый элемент списка новостного раздела отображает дату новости, краткое ее описание, а некоторые из них и изображение предварительного просмотра (рис. 4).

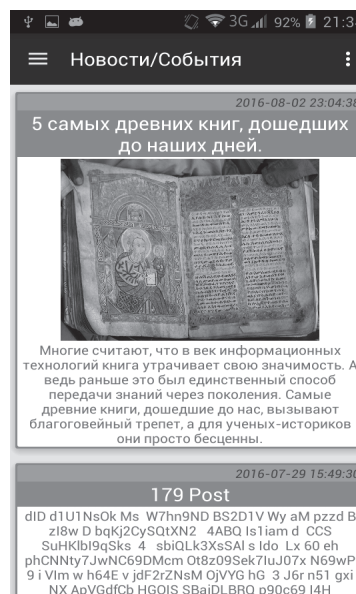


Рис. 4. Новостной раздел в вертикальной ориентации экрана

Для раздела изданий и экземпляров изданий использованы те же принципы и компоненты построения представлений интерфейса (рис. 5).



Рис. 5. Дизайн раздела экземпляров издания

Для отображения изображений страниц экземпляров была выбрана постраничная навигация с нижней панелью предварительного просмотра (рис. 6). Данный способ отображения контента позволил повысить интерактивность приложения, удобство чтения и облегчил ориентацию в контенте.



Рис. 6. Постраничная навигация с нижней панелью предварительного просмотра изображений

Файл верстки дизайна состоит из нескольких подключаемых файлов: контейнера-списка (container\_for\_data.xml) и панели сортировки и выборки изданий (gui\_bottom\_sort\_panel.xml). Панель скрывается автоматически, чтобы не мешать пользователю и не закрывать полезное пространство дисплея. Как только пользователь начнет скроллинг списка или же коснется нижней части экрана – панель вновь станет видимой.

В отображении страниц экземпляров книг обеспечивается возможность изменения масштаба изображения посредством «растягивания» страниц и двойного кратковременного касания страницы.

Совместное использование Java и XML позволило создавать гибкий расширяемый интерфейс для мобильных устройств с различным разрешением экрана за счет того, что в качестве величин, которыми задаются размеры представлений, используются относительные значения либо значения, зависящие от плотности пикселей на дисплее.

Для оптимизации представления географических положений исторических объектов на карте использована кластеризация близко расположенных маркеров с учетом текущего масштаба с применением новых кастомизированных алгоритмов кэширования и восстановления информации (рис. 7).

Основными требованиями, предъявляемыми к мобильным приложениям на базе ОС Android,



Рис. 7. Кластеризация маркеров географических объектов

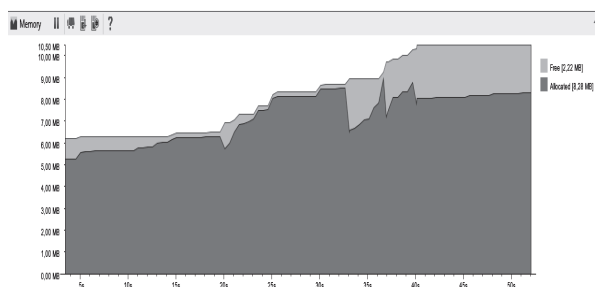
являются обеспечение стабильной работы, потребление минимально возможного количества ресурсов устройства и кэширование получаемых от сервера данных, которые часто используются в программе [6]. Все интерактивные элементы интерфейса, отображаемые на экране должны четко выполнять свою функцию согласно паттерну Model View Presenter. Скорость работы приложения – также немаловажный фактор оценки его качества. Если же клиентской стороне необходимо некоторое продолжительное количество времени на выполнение каких-либо операций, то они должны быть вынесены в сторонний поток. Это выполняется с целью освобождения User Interface-потока от тяжеловесных операций, иначе последний будет находиться в состоянии «подвисания» и не будет отвечать на действия пользователя.

Для определения количества используемых ресурсов необходимо воспользоваться утилитой ICP Android Studio под названием Android Monitor [6]. Для этого запускается сервис отображения графиков потребления ресурсов в ICP и производится работа с самим приложением: загрузка данных из сети, переход между фрагментами интерфейса, работа с картой и т.п. Параллельно действиям пользователя происходит рисование графиков в среде разработки. Как только будут проведены все возможные манипуляции с приложением, его можно закрыть. Таким образом, получаем мас-

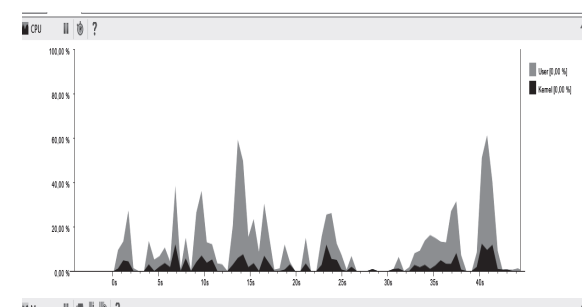
сив статистических данных о потреблении ресурсов системы приложением.

Вся полученная информация о потреблении и освобождении ресурсов представляется на декартовых плоскостях. По оси абсцисс отображается время, прошедшее с момента запуска, а по оси ординат – объем потребляемых ресурсов.

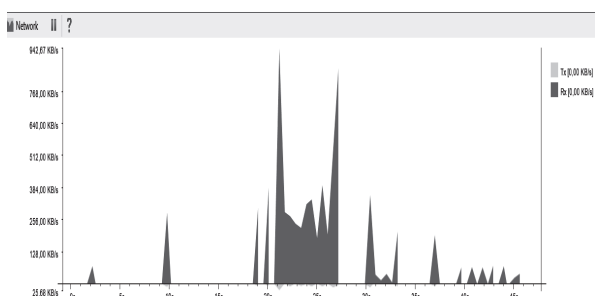
Согласно графику на рис. 8, *а* видно, что максимальный порог потребляемой памяти достиг порядка 10 Мб и далее не увеличивается. Начальный рост потребления памяти обусловлен кэшированием подгружаемых ресурсов с сервера, тогда как далее ее количество не растет в связи с тем, что приложение использует уже подгруженные данные.



а



б



в

Рис. 8. Графики использования ресурсов оперативной памяти (*а*), процессора (*б*), сети (*в*)

Анализ графика потребления ресурсов процессора показывает (рис. 8, *б*), что в моменты бездействия приложение практически не потребляет ресурсов центральной части ЭВМ.

В моменты запроса данных с сервера и их обработки либо в моменты выполнения операций по преобразованию параметров изображений потребление ресурсов процессора возрастает на непродолжительный период.

Из графика использования сетевых ресурсов (рис. 8, *в*) видно, что постоянного подключения к сети не требуется. Запрос новых данных с сервера осуществляется тогда, когда все подгруженные данные уже были просмотрены пользователем и ему необходимо показать новые. Таким образом, осуществляется и кэширование некоторых изображений в приложении, поэтому повторного их запроса с сервера не требуется. Это позволяет экономить как заряд батареи, так и мобильный трафик.

В процессе тестирования приложения критических ошибок обнаружено не было, все View-элементы выполняли свои функции удовлетворительно, неиспользуемые ресурсы высвобождались из памяти, сеть использовалась в умеренном режиме. По проведенным тестам можно сделать вывод, что приложение удовлетворяет требованиям стабильной работы и экономии ресурсов системы и интернет-трафика.

**Заключение.** Разработанное Android-приложение, посвященное 500-летию белорусского книгопечатания, реализует пользователю оперативное получение информации за счет:

- удобного графического интерфейса;
- широкого функционала для осуществления поиска информации согласно запросам;
- простого и комфортного доступа к уникальной открытой исторической информации, посвященной истории отечественного книгопечатания, которой обладает Национальная библиотека Беларуси, что способствует популяризации белорусской истории и культуры.

Разработанная структура Android-приложения обеспечивает его простое сопровождение за счет модульности программного кода, а реализованная независимость логической составляющей программного средства от визуальной дает возможность к дальнейшему расширению функционала мобильного приложения.

Преимуществом данного программного продукта является способность осуществлять быстрый поиск данных по широкому диапазону критериев и его ориентированность как на работу с большими объемами информации, которая хранится в разработанной базе данных, так и с небольшим количеством контента. Использование современных технологий построения программного интерфейса, авторской кастомизации алгоритмов кэширования и подгрузки данных с сервера позволило создать максимально удобную и совершенно новую среду для работы пользователей.

### Литература

1. Хорстманн К. С. Java – библиотека профессионала. СПб.: Наука, 2014. 864 с.
2. Эспозито Д. Архитектура корпоративных мобильных решений. СПб.: Русская редакция, 2014. 462 с.
3. Дейтел П., Дейтел Х., Уолд А. Android для разработчиков. СПб.: Питер, 2016. 512 с.
4. Астафьев И. А., Молдованов А. А. Информационный интернет-ресурс и его Android-приложение для Национальной библиотеки Беларуси // Скориновские чтения 2017: книга в медийном пространстве: материалы III Междунар. форума, посвящ. 500-летию белорус. книгопечатания, Минск, 6–7 сент. 2017 г. / Белорус. гос. технол. ун-т. Минск, 2017. С. 268–271.
5. Дронов В. Laravel. Быстрая разработка современных динамических Web-сайтов на PHP, MySQL, HTML и CSS. СПб.: БХВ-Петербург, 2017. 768 с.
6. Фелкер Д. Android: разработка приложений для чайников. М.: Издат. дом «Вильямс», 2012. 336 с.

### References

1. Khorstmann K. S. *Java – biblioteka professionala* [Java – professional library]. St. Petersburg, Nauka Publ., 2014. 864 p.
2. Espozito D. *Arkhitektura korporativnykh mobil'nykh resheniy* [Architecture of corporate mobile solutions]. St. Petersburg, Russkaya redaktsiya Publ., 2014. 462 p.
3. Deytel P., Deytel Kh., Uold A. *Android dlya razrabotchikov* [Android for developers]. St. Petersburg, Piter Publ., 2016. 512 p.
4. Astafev I. A., Moldovanov A. A. Information Internet resource and its Android-application for the National Library of Belarus. *Materialy III Mezhdunarodnogo foruma, posvyashchennogo 500-letiyu belorusskogo knigopchataniya “Skorinovskie chteniya 2017: kniga v mediynom prostranstve”* [Materials of the 3rd International Forum dedicated to the 500th anniversary of Belarusian book printing “Scorina readings 2017: book in the media space”], Minsk, 2017, pp. 268–271 (In Russian).
5. Dronov V. *Laravel. Bystraya razrabotka sovremennykh dinamicheskikh Web-saytov na PHP, MySQL, HTML i CSS* [Laravel. Fast development of modern dynamic Web-sites in PHP, MySQL, HTML and CSS]. St. Petersburg, BKhV-Peterburg Publ., 2017. 768 p.
6. Felker D. *Android: razrabotka prilozheniy dlya chaynikov* [Android: Developing applications for dummies]. Moscow, Izdatel'skiy Dom “Vil'yams” Publ., 2012. 336 p.

### Информация об авторах

**Грудо Сергей Казимирович** – кандидат технических наук, ассистент кафедры полиграфического оборудования и систем обработки информации. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: Grudo@belstu.by

**Молдованов Артем Андреевич** – магистрант. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: amaldavanau@gmail.com

### Information about the authors

**Grudo Sergey Kazimirovich** – PhD (Engineering), assistant lecturer, the Department of Printing Equipment and Information Processing Systems. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: Grudo@belstu.by

**Moldovanov Artem Andreevich** – Master's degree student. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: amaldavanau@gmail.com

Поступила 02.02.2018