

## МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ МОНИТОРИНГА, ДИАГНОСТИКИ И ПРОГНОЗИРОВАНИЯ РИСКА ОТКАЗОВ КОМПОНЕНТОВ СЛОЖНОЙ ТЕХНИЧЕСКОЙ СИСТЕМЫ

В.В. Вычужанин, Н.Д. Рудниченко, А.В. Вычужанин

Одесский национальный политехнический университет,  
просп. Шевченко, 1, Одесса, 65044, Украина; e-mail: 126.ist.onpu@gmail.com

Проведенный анализ технических решений, позволяющих повысить надежность функционирования сложных технических систем, показал, что своевременная и качественная диагностика, в том числе дистанционная, компонентов сложных технических систем при их эксплуатации позволяет повысить надежность систем и эффективность их эксплуатации. Одним из важнейших показателей надежности является оценка риска отказов компонентов сложных технических систем. Программная и аппаратная беспроводная передача данных в информационных системах со смартфонами позволяет осуществлять дистанционное управление и контроль расходов ресурсов, синхронизацию работы компонентов сложных технических систем, координацию распределенных вычислительных процессов. В статье рассматриваются аспекты разработки мобильных приложений для дистанционного мониторинга, диагностики и прогнозирования риска отказов компонентов сложной технической системы. Описаны характеристики дизайна пользовательского интерфейса и порядок взаимодействия с разработанным приложением для дистанционного мониторинга, диагностики и прогнозирования риска отказов компонентов сложной технической системы. Для реализации приложения разработана логическая модель базы данных, основанная на выборе системы управления базой данных. В статье приведены результаты использования диаграммы классов, а также проект мобильного приложения с обобщенным алгоритмом. Приведена модульная структура реализации программного обеспечения мобильного приложения для мониторинга, диагностики и прогнозирования риска отказов компонентов сложной технической системы. Разработанный проект мобильного приложения для мониторинга, диагностики и прогнозирования риска отказов компонентов сложной технической системы является полным и логическим завершением. В процессе написания кода программного обеспечения рекомендуется использовать настройки Gradle, что позволит ускорить процесс рефакторинга, профилирования и интеграции с системой контроля версий GIT. Разработанное мобильное приложение позволяет упростить процесс оценок риска компонентов сложных технических систем. Дополнительный контент и функциональные дополнения проекта могут быть возможны благодаря интерфейсам Android-Core. В качестве альтернативы серверной стороне рекомендуется использовать современные облачные сервисы и технологии, основанные на моделях IaaS и PaaS.

**Ключевые слова:** мобильное приложение, мониторинг, диагностика, прогнозирование, андроид, технические системы, пользовательский интерфейс, Java, прототипирование.

### Введение

Проведение своевременной диагностики, в том числе дистанционной сложных технических систем (СТС) позволяет повысить надежность систем, а значит и эффективность их эксплуатации. Одним из показателей надежности СТС является оценка риска отказов компонентов систем [1-5]. Переход к обеспечению эксплуатационной надежности технологического оборудования по его «фактическому состоянию» вызывает необходимость предпринимать меры по предупреждению и

обнаружению причин возникновения отказов СТС, создания систем мониторинга, диагностики и прогнозирования их состояния [6-10]. Решение проблемы возможно использованием информационных систем дистанционного мониторинга, диагностики и прогнозирования состояния компонентов СТС.

Развивающиеся мобильные операционные системы и технологии находят все более широкое применение на базе мобильных гаджетов для решения различных задач [11-13]. Программная и аппаратная беспроводная передача данных в информационных системах со смартфонами позволяет осуществлять дистанционное управление и контроль расходов ресурсов, синхронизацию работы компонентов сложных технических систем, координацию распределенных вычислительных процессов [14,15]. Следует также отметить актуальность использования открытой операционной системы Android в качестве платформы для разработки мобильных приложений, которая распространяется на лицензиях GPU и Apache [16]. Система обладает преимуществами: поддержка интеграции сторонних сервисов и компонентов; наличие механизмов реализации для виртуализации; гибкость реализации приложений в Java-шаблонах MVC и шаблонах проектирования; поддержка защиты протокола SSL-протокола от передаваемой информации; возможность оптимизации передачи мобильного трафика данных [17]. Поддерживаемая функциональность платформы Android позволяет разрабатывать мобильные приложения для удаленного мониторинга и прогнозирования риска отказов технических компонентов системы [18,19].

## Цель работы

*Целью* работы является разработка проекта мобильного приложения для дистанционного мониторинга, диагностики и прогнозирования риска отказов компонентов СТС, включающего: обоснование специфики предложения пользовательского интерфейса и опыта взаимодействия с приложениями мобильного удаленного мониторинга; разработку логической модели базы данных для мобильных приложений; проектирование UML-диаграмм для создания классов абстрактных моделей и объектов мобильных приложений; алгоритм мобильного приложения; модульную структуру; интерфейс прототипа на главном экране.

## Основная часть

Разработка мобильного приложения.

А. Специфика пользовательского интерфейса и опыт взаимодействия мобильного приложения.

Разрабатываемое мобильное приложение предназначено для использования на мобильных устройствах с версией операционной системы Android 4.4.2 и выше, размер экрана 4,5 дюйма и разрешение 800\*600 пикселей и более. Гибкость интерфейса пользователя достигается за счет: элементов цветodelения взаимодействия и визуализации действия динамики при их активации; возможности изменения размера и шрифта текстовых блоков и типов надписей с помощью Typeface; поддержки переключения между вкладками экрана с использованием метода выбора типа обработки событий; преобладания плоских элементов дизайна над skeuomorph; ясности и динамической анимации при рендеринге статистики в графической форме; частичного размытия фоновой активности с появлением диалоговых окон или информационных сообщений; размещения всех функциональных элементов в одном экране, исключая необходимость прокрутки вверх/вниз; интегрированной интеллектуальной клавиатуры для ввода текстовых данных.

В. Разработка модели логической базы данных.

Выбор системы управления базами данных (СУБД) для разработки и внедрения базы данных эффективно взаимодействующей с мобильным клиентским приложением усложняется из-за обилия на рынке доступных решений. СУБД SQLite - встроенная поддержка файлового сервера ОС Android является преимуществом в случае полностью автономного режима. Операции мобильного приложения связаны с необходимостью постоянного подключения к удаленному внешнему серверу. Использование СУБД приемлемо из-за интеграции стандартных инструментов и поддержки ее библиотеки, что позволяет увеличить скорость и эффективность работы мобильного приложения. В долгосрочной перспективе при разработке проекта на основе новых списков требований для соответствующей масштабируемости может быть выбрана NoSQL СУБД MongoDB.

Для решения поставленной задачи разработаны таблицы базы данных: элементы, межэлементные коммуникационные подсистемы, системы, датчики, параметры, вероятность отказов компонентов СТС, ущерб, параметры прогноза, список журналов. Типы данных, используемых в базе данных - в основном целые действительные числа, прописные и регистрационные данные в таблице. На основе разработанной ER-модели может быть реализована конкретная физическая модель базы данных в MySQL Workbench или SQL Navigator.

### С. Разработка UML-диаграмм мобильного приложения

Проектирование сценариев использования приложения может быть обеспечено диаграммой вариантов использования мобильных приложений (рис. 1).

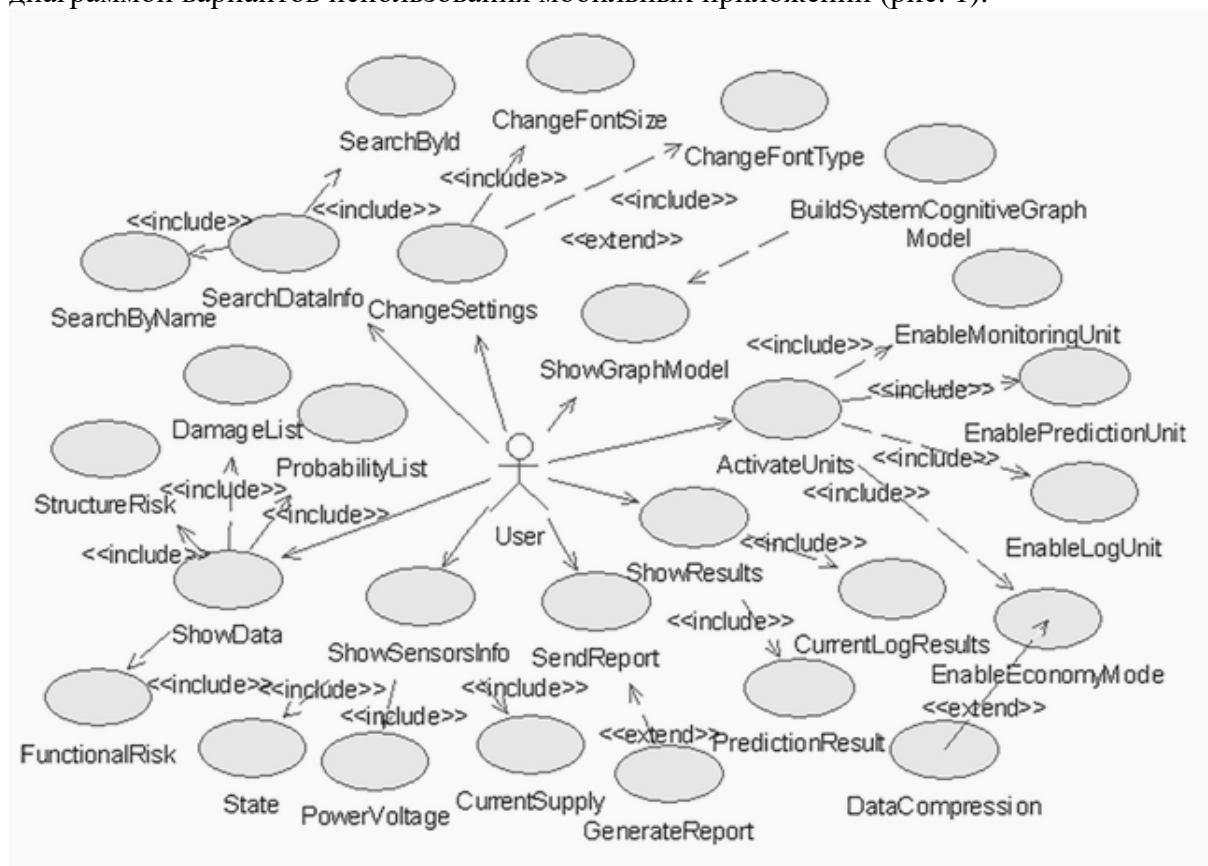


Рис. 1. Диаграмма вариантов использования мобильных приложений

Были выбраны функции приложений: просмотр информации о повреждениях компонентов СТС; оценка вероятности сбоя, структурных и функциональных рисков; поиск информации в базе данных по имени или уникальному номеру объекта; локальное хранилище и отправка сгенерированного отчета в формате pdf; построение когнитивной имитационной модели (КИМ) [6]; изменение настроек пользовательского интерфейса; просмотр результатов прогноза состояния компонентов СТС; включение и

отключение модулей мониторинга, диагностики и прогнозирования состояния компонентов СТС; ведение журнала и переход в экономичный режим, уменьшающий энергопотребление батареи мобильного устройства, а также обеспечивающий дополнительное сжатие данных, отправленных на сервер.

Для формализации класса и объектных моделей мобильного приложения разработана диаграмма классов проекта, а также указаны отношения между классами и их экземплярами (рис. 2).

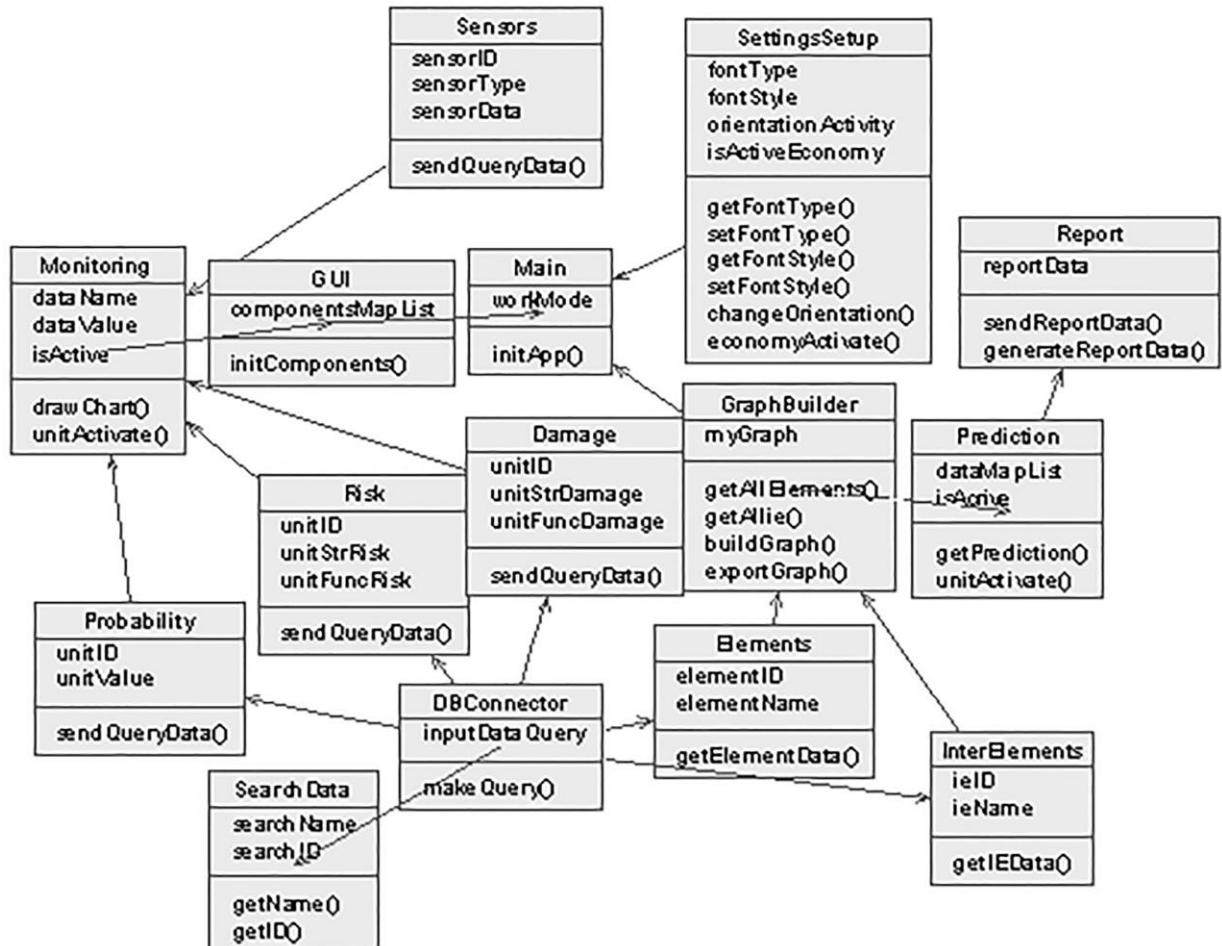


Рис. 2. Фрагмент диаграммы класса мобильного приложения

Запуск приложения выполняется в отдельном потоке с помощью метода `initApp` Main class. Процедура авторизации выполняется в классе приложений `Authoriz` и создает графическую активность пользователя с полями входа и пароля. Каждый класс реализует: мониторинг, диагностику, прогнозирование; построение графика КИМ; поиск и просмотр информации с датчиков контролируемых параметров СТС; определение вероятности отказов компонентов СТС, риска и ущерба. Для более подробного описания мобильного приложения была разработана диаграмма его деятельности (рис. 3). Объекты на диаграмме: клиент-мобильное приложение; внешний сервер, синхронизирующий, обрабатывающий и проверяющий статистические данные о работе компонентов СТС; сервер управления - выполняет задачи хранения, обработки, резервирования и обмена данными с внешним сервером и системой сбора данных; система сбора данных - выполняет функции сбора данных непосредственно с датчиков и передает информацию на сервер управления СТС. Для проверки активности сервера и возможности установления соединения между сервером и клиентом используется мобильное приложение. Оно отправляет пакет запросов для проверки основных активных обновлений в репозиториях, результатов проверки

данных авторизации, а также получает техническую и статистическую информацию о работе СТС.

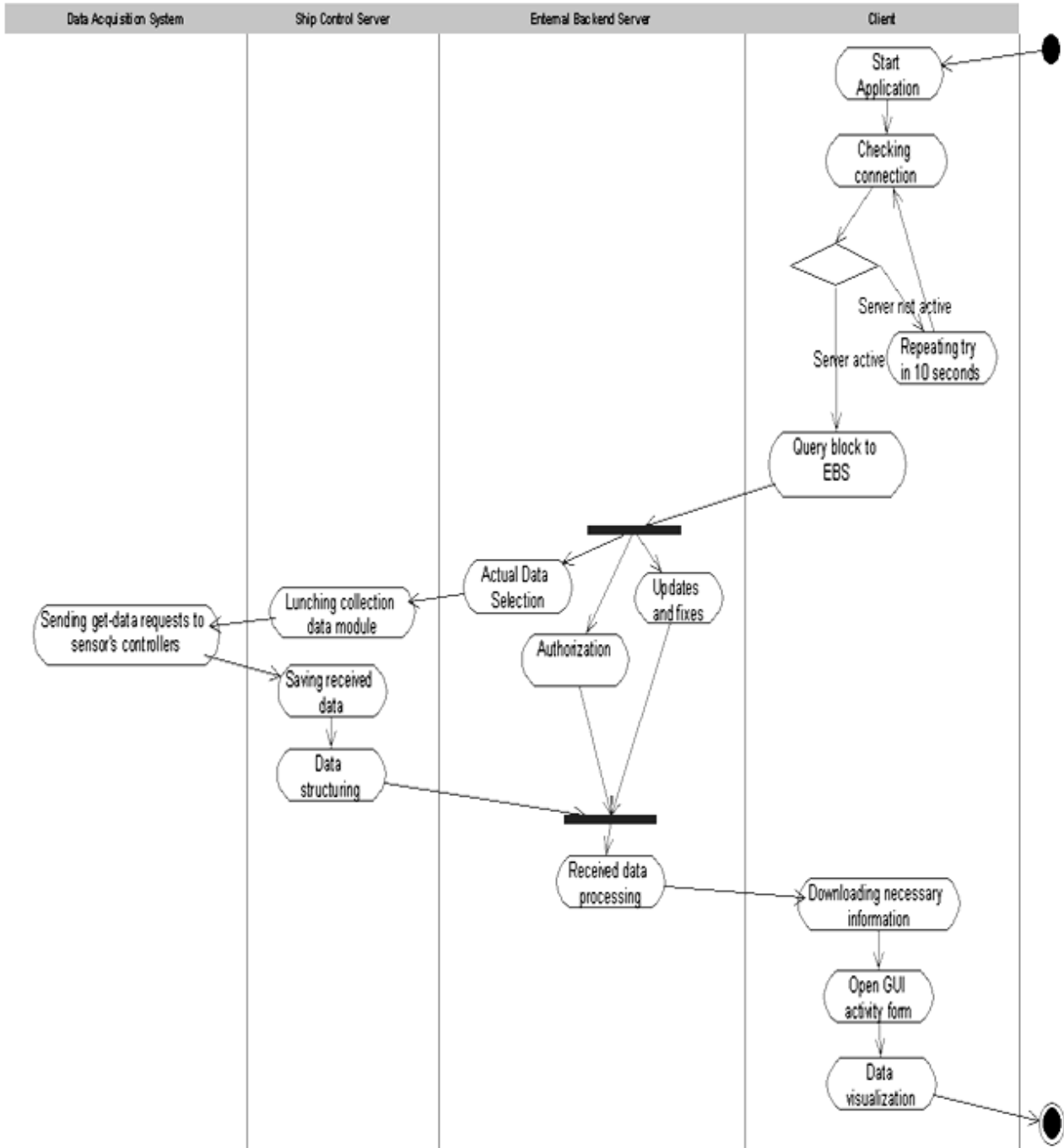


Рис. 3. Диаграмма активности мобильного приложения

D. Алгоритм мобильного приложения.

Для реализации интерфейса прототипа и написания программной кодовой реализации, формализованной через UML-функциональность разработан алгоритм работы мобильного приложения (рис. 4). Пакет установки приложения загружается на мобильное устройство в формате \*.apk. В результате выполняется инициализация всех компонентов и зависимостей программных приложений, включая проверки подключения к точкам беспроводного доступа в Интернет с помощью поддерживаемых технологий (с использованием пакета android.net) и действия удаленного сервера.

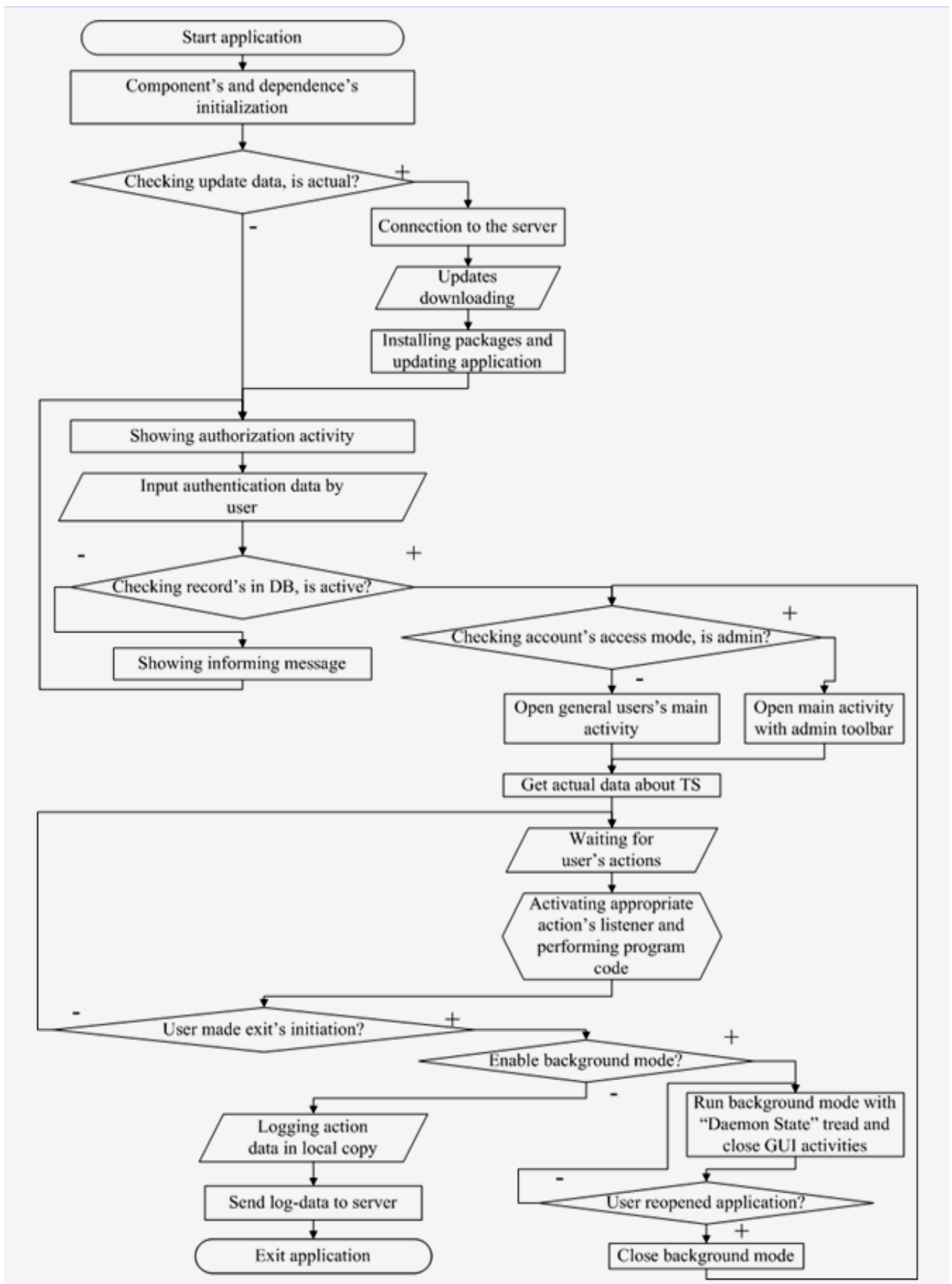


Рис. 4. Алгоритм мобильного приложения

Затем осуществляется процесс визуализации компонента отражения обновления данных на сервер. В случае реализации такой процедуры новый процесс начинает установку необходимых пакетов и обновление приложений в фоновом режиме. Пользователь применяет интерфейс входа в систему, введя имя пользователя и пароль.

Осуществляется получение технической и статистической информации о компонентах системы. После этого приложение переходит в основную форму, в режим ожидания и режим пользовательских запросов, в соответствии с которым осуществляется выполнение программного кода, содержащегося в соответствующем обработчике событий. При закрытии приложения появляется визуализация дополнительного диалогового окна, в котором предлагается переносить работу мобильных приложений в фоновом режиме. Если пользователь выбирает запуск приложения в фоновом режиме в отдельном процессе и потоке, тогда действие GUI выгружается из оперативной памяти мобильного устройства. Если снова запускается приложение, которое в этой точке уже находится в фоновом режиме, тогда режим закрывается, и управление передается для проверки доступа к режиму приложения. В случае если пользователь инициирует окончательное закрытие приложения в фоновом режиме, происходит ведение журнала всех действий, выполняемых в локальной копии рабочего каталога мобильного приложения. При появлении соединения с сервером, то выполняется отправка данных на сервер. После этого происходит полная разрядка приложения из основной памяти используемого мобильного устройства. Объем данных мобильных приложений, хранящихся в соответствующем каталоге «Cache», не должен превышать 2,5 мегабайта. В противном случае произойдет процедура кэширования.

Е. Модульная структура приложения.

Разработанное мобильное приложение состоит из модулей:

- инициализации компонентов пользовательского интерфейса;
- проверки текущего модуля данных программы для версии приложения;
- подключения к удаленному серверному модулю;
- построения и визуализации системного модуля КИМ [20];
- отчетности, для преобразования, экспорта статистики и графических данных из приложений;
- прогноза для создания и обучения искусственной нейронной сети на основе метода обратного распространения, линейной функции нормализации с помощью касательной функции активации. Создается нейронная сеть, обучаемая с оценкой результатов, в виде значений ошибок [6];
- визуализации статистических данных;
- получения данных для реализации запросов выборки данных, хранящихся в базе данных на стороне сервера.

Модули представляют собой базовую структуру проекта разработки мобильных приложений. Они хранятся в отдельных пакетах, поэтому в будущем проект может быть дополнен. Прототип реализации программного обеспечения Интерфейс клиента имеет форму (рис. 5).

Прототип разработан с использованием облачной службы SaaS fluidui.com, состоящей из 3 вкладок: Analytics (содержит компоненты, просматривающие список активных датчиков, получения информации и управления ими модулями мониторинга, диагностики и прогнозирования, ведения журналов), мониторинг (включает графические компоненты динамической визуализации параметров и системы характеристик), Prediction (содержит таблицу прогнозируемых значений риска в зависимости от выбранного периода времени).

При реализации прототипа мобильного клиента проведено профилирование его функционирования с целью исследования специфики загрузки имплементированных компонентов, обработки и отправки данных через установленное с сервером соединение, а также осуществления оценки общей производительности приложения.

Исследование осуществлялось под тремя различными версиями Android в двух режимах: стандартном и экономном (сжатие передаваемых данных и сокращение вычислительных операций), оценивался объем занимаемой оперативной памяти, ресурс

процессора и объем передаваемых данных за одну итерацию обращения к серверу, результаты приведены в таблице 1.

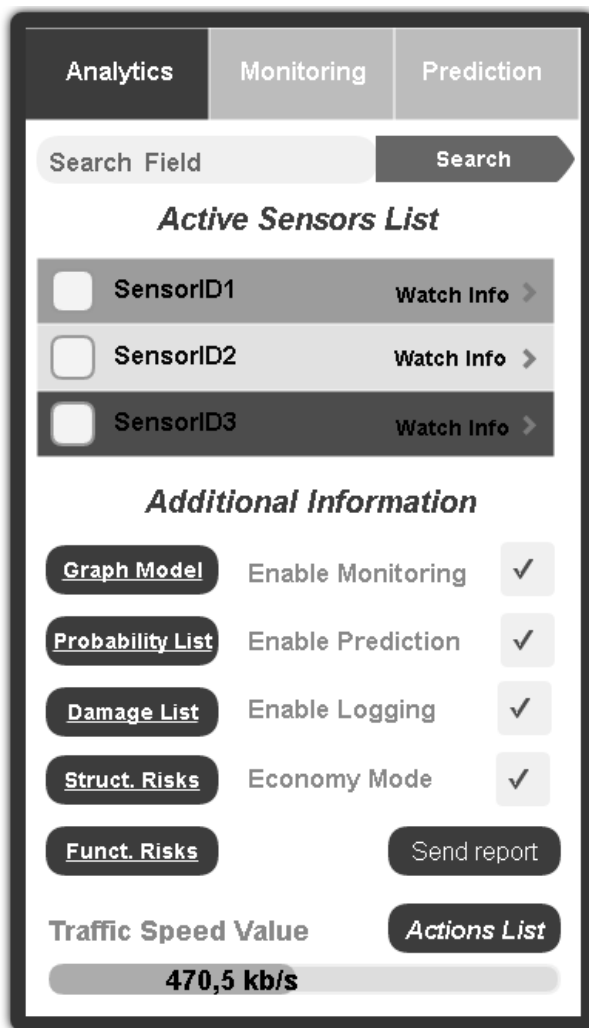


Рис. 5. Прототип интерфейса мобильного клиента

Таблица 1.

Результаты профилирования мобильного приложения

	Стандартный режим			Экономный режим		
	RAM, mb	CPU, %	Data, mb	RAM, mb	CPU, %	Data, mb
Android 5.x	65	25-30	4,5	59	25-30	4,0
Android 6.x	59	20-30	6,7	55	22-27	6,0
Android 7.x	77	15-20	7,1	58	15-18	6,8

Следует отметить, что использование оперативной памяти в стандартном режиме эксплуатации приложения является максимальным в операционной системе Android 7.x, что связано с необходимостью загрузки дополнительных компонентов для обеспечения работы ядра системы. При этом, процессор мобильного устройства задействован для проведения вычислительных операций в среднем на 5-10% меньше, чем в случаях использования предыдущих версий Android, что свидетельствует о лучшей согласованности в использовании API Android 7.x. Объемы передаваемых данных в различных режимах использования изменяются не существенно, что является основанием для дальнейшего совершенствования алгоритмов их сжатия. Экономичный



режим использования приложения под управлением Android 7.x в большей степени позволяет снизить объем используемой оперативной памяти, чем степень загрузки процессора, в более ранних версиях операционной системы эта зависимость менее заметна, отличия составляет 4-6% для Android 6.x и 5.x соответственно.

Проведенные исследования обуславливают целесообразность использования разработанного мобильного приложения под управлением операционной системы Android 7.x в связи с лучшей организацией процессов обмена данными и оптимизацией распределения ресурсов для проведения вычислений.

## Выводы

Разработанный проект мобильных приложений для мониторинга, диагностики и прогнозирования риска отказов компонентов сложной технической системы является полным и логическим завершением. В процессе написания кода программного обеспечения целесообразно использование настройки Gradle, что позволит ускорить процесс рефакторинга, профилирования и интеграции с системой контроля версий GIT. Разработанное мобильное приложение позволяет упростить процесс оценки риска отказов компонентов сложных технических систем. Дополнительный контент и функциональные дополнения проекта могут быть возможны благодаря интерфейсам Android-Core. В качестве альтернативы серверной стороне рекомендуется использовать современные облачные сервисы и технологии, основанные на моделях IaaS и PaaS.

## Список литературы

1. Andersen, B. A Diagnostic System for Remote Real-Time Monitoring of Marine Diesel-Electric Propulsion Systems / B. Andersen. – Leipzig, 2011. – 45 p.
2. O'Neill, J. Technical Risk Assessment: a Practitioner's Guide / J. O'Neill, N. Thakur, A. Duus. – Australia, 2007. – 29 p.
3. Kertzner, P. Process Control System Security Technical Risk Assessment Methodology & Technical Implementation / P. Kertzner, J. Watters, D. Bodeau // Research Report. – 2008. – No. 13. – 47 p.
4. Вычужанин, В.В. Математические модели нестационарных режимов воздухообработки в центральной СКВ / В.В. Вычужанин // Вісник Одеського національного морського університету, збірник наукових праць, 2014. – №23. – С. 172-186.
5. Вычужанин, В.В. Оценки структурного и функционального рисков сложных технических систем / В.В. Вычужанин, Н.Д. Рудниченко // Восточно-Европейский журнал передовых технологий. – 2014. – Том 1, № 2(67). – С. 18-22.
6. Vychuzhanin, V. Devising a method for the estimation and prediction of technical condition of ship complex systems / V. Vychuzhanin, N. Rudnichenko, V. Boyko, N. Shibaeva // Eastern-European Journal of Enterprise Technologies. – 2016. – No. 6/9. – Pp. 4-11.
7. Vychuzhanin, V. Assessment of risks structurally and functionally complex technical systems / V.V. Vychuzhanin, N.D. Rudnichenko // Eastern-European Journal of Enterprise Technologies. – 2014. – No. 2. – Pp. 18-22.
8. Вычужанин, В.В. Технические риски сложных комплексов функционально взаимосвязанных структурных компонентов судовых энергетических установок / В.В. Вычужанин, Н.Д. Рудниченко // Вісник Одеського національного морського університету, збірник наукових праць. – 2014. – № 2(40). – С. 68-77.
9. Вычужанин, В.В. Информационное обеспечение мониторинга и диагностирования технического состояния судовых энергоустановок / В.В. Вычужанин // Вісник одеського національного морського університету, збірник наукових праць. – 2012. – № 35. – С. 111-124.
10. Вычужанин, В.В. Метод управления рисками судовых сложных технических систем / В.В. Вычужанин, Н.Д. Рудниченко // Проблеми техніки. – 2014. – №2. – С. 138-142.
11. Аксенов, К.В. Обзор современных средств для разработки мобильных приложений / К.В. Аксенов // Новые информационные технологии в автоматизированных системах. – 2014. – № 17. – С. 508-513.

12. Голощанов, А.Л. Google Android: программирование для мобильных устройств / А.Л. Голощанов. – СПб.: БХВ-Петербург, 2011. – 448 с.
13. Майорова, Е.С. Современное состояние средств разработки мобильных приложений на платформах iOS, Android и Windows Phone / Е.С. Майорова, В.А. Ошурков, Л.Е. Цуприк // Перспективы науки и образования, 2015. – № 4(16). – С. 83-87.
14. Терехов, А.Н. Технология разработки мобильных онлайн сервисов / А.Н. Терехов, В.В. Оносовский // Конференция СЕЕ-SECR, 2011. – С. 1-2.
15. Соколова, В.В. Разработка мобильных приложений / В.В. Соколова. – Томск: Изд-во Томского политехнического университета, 2011. – 175 с.
16. Ableson, W. Android in action / W. Ableson, R. Sen, C. King. – Manning Publications, 2011. – 592 p.
17. Burnette, E. Hello, Android: introducing Google's mobile development platform / E. Burnette. – Pragmatic Bookshelf, 2010. – 300 p.
18. Fling, B. Mobile design and development: practical concepts and techniques for creating mobile sites and web apps / B. Fling. – O'Reilly Media, 2009. – 336 p.
19. To, N. The Android developer's cookbook: building applications with the Android SDK (Developer's Library) / N. To, J. Steele. – Addison-Wesley Professional, 2010. – 400 p.
20. Вычужанин, В.В. Информационная когнитивная модель технологической взаимозависимости сложных технических систем / В.В. Вычужанин, Н.Д. Рудниченко // Информатика и математические методы в моделировании. – 2013. – № 3. – С. 240-247.

### **МОБІЛЬНИЙ ДОДАТОК ДЛЯ МОНІТОРИНГУ, ДІАГНОСТИКИ ТА ПРОГНОЗУВАННЯ РИЗИКУ ВІДМОВ КОМПОНЕНТІВ СКЛАДНОЇ ТЕХНІЧНОЇ СИСТЕМИ**

В.В. Вичужанин, Н.Д. Рудніченко, О.В. Вичужанин

Одеський національний політехнічний університет,  
проспект Шевченка, 1, Одеса, 65044, Україна; e-mail: 126.ist.onpu@gmail.com

Проведений аналіз технічних рішень, що дозволяють підвищити надійність функціонування складних технічних систем показав, що своєчасна і якісна діагностика, в тому числі дистанційна компонентів складних технічних систем при їх експлуатації дозволяє підвищити надійність систем і ефективність їх експлуатації. Одним із найважливіших показників надійності є оцінка ризиків компонентів складних технічних систем. Програмна та апаратна бездротова передача даних в інформаційних системах зі смартфонами дозволяє здійснювати дистанційне керування і контроль витрат ресурсів, синхронізацію роботи компонентів складних технічних систем, координацію розподілених обчислювальних процесів. У статті розглядаються аспекти розробки мобільних додатків для дистанційного моніторингу, діагностики та прогнозування ризику відмов компонентів складної технічної системи. Описано характеристики дизайну користувальницького інтерфейсу і порядок взаємодії з розробленим додатком для дистанційного моніторингу, діагностики та прогнозування ризику відмов компонентів складної технічної системи. Для реалізації програми розроблена логічна модель бази даних, заснована на виборі системи управління базою даних. У статті наведені результати використання діаграми класів, а також проект мобільного додатка з узагальненим алгоритмом. Наведено модульну структуру реалізації програмного забезпечення мобільного застосування для моніторингу, діагностики та прогнозування ризику відмов компонентів складної технічної системи. Розроблений проект мобільного застосування для моніторингу, діагностики та прогнозування ризику відмов компонентів складної технічної системи є повним і логічним завершенням. В процесі написання коду програмного забезпечення рекомендується використовувати налаштування Gradle, що дозволить прискорити процес рефакторинга, профілювання і інтеграції з системою контролю версій GIT. Розроблене мобільний додаток дозволяє спростити процес оцінок ризику компонентів складних технічних систем. Додатковий контент і функціональні доповнення проекту можуть бути можливими завдяки інтерфейсів Android-Core. В якості альтернативи серверній стороні рекомендується використовувати сучасні хмарні сервіси і технології, засновані на моделях IaaS і PaaS.

**Ключові слова:** мобільний додаток, моніторинг, діагностика, прогнозування, андроїд, технічні системи, призначений для користувача інтерфейс, Java, прототипування.

**MOBILE APPENDIX FOR MONITORING, DIAGNOSTICS AND FORECASTING  
RISK OF FAILURE OF COMPONENTS OF COMPLEX TECHNICAL SYSTEM**

V.V. Vychuzhanin, N.D. Rudnichenko, A.V. Vychuzhanin

Odessa National Polytechnic University,  
Shevchenko Avenue, 1, Odessa, 65044, Ukraine  
e-mail: 126.ist.onpu@gmail.com

The analysis of technical solutions allowing to improve the reliability of functioning of complex technical systems has shown that timely and high-quality diagnostics, including remote components of complex technical systems during their operation, can improve the reliability of systems and the efficiency of their operation. One of their most important indicators of reliability is the risk assessment of components of complex technical systems. Software and hardware wireless data transmission in information systems with smartphones allows remote control and monitoring of resource costs, synchronization of components of complex technical systems, coordination of distributed computing processes. The article deals with the development of mobile applications for remote monitoring, diagnostics and forecasting the risk of failures of components of a complex technical system. Describes the characteristics of the design of the user interface and the order of interaction with the developed application for remote monitoring, diagnosis and prediction of the risk of failures of components of a complex technical system. To implement the application, a logical model of the database was developed, based on the choice of a database management system. The article shows the results of using a class diagram, as well as a draft mobile application with a generalized algorithm. The modular structure of software implementation for a mobile application for monitoring, diagnosing and forecasting the risk of failures of components of a complex technical system is given. The developed project of a mobile application for monitoring, diagnosing and forecasting the risk of failures of components of a complex technical system is a complete and logical conclusion. In the process of writing software code, it is recommended to use the Gradle settings, which will speed up the process of refactoring, profiling and integration with the GIT version control system. The developed mobile application allows to simplify the process of risk assessments of components of complex technical systems. Additional content and functional additions to the project may be possible thanks to the Android-Core interfaces. As an alternative to the server side, it is recommended to use modern cloud services and technologies based on the IaaS and PaaS models.

**Keywords:** mobile application, monitoring, diagnostics, forecasting, android, technical systems, user interface, Java, prototyping.