

УДК 004.7

DOI: 10.25559/SITITO.16.202001.118-126

Система информирования о чрезвычайных ситуациях на основе ad hoc сетей

Т. А. Приходько, Н. А. Жудин*, С. А. Воробьев

Кубанский государственный университет, г. Краснодар, Россия

350040, Россия, г. Краснодар, ул. Ставропольская, д. 149

* dantes04015380209@gmail.com

Аннотация

Одной из современных тенденций организации устойчивой к задержкам связи в условиях высокодинамичной маршрутизации является формирование самоорганизующихся оппортунистических сетей с использованием интеллектуальных мобильных устройств. Современные смартфоны скрывают большой потенциал для мониторинга и управления чрезвычайными ситуациями: они широко распространены, они могут устанавливать одноранговые беспроводные линии связи с использованием технологий связи на коротких расстояниях, гарантируя таким образом охват, даже когда стационарные инфраструктуры недоступны, они могут с высокой точностью определять свое местоположение с помощью нескольких встроенных датчиков, и отправлять сообщения при отсутствии глобальной связи.

В работе изучаются возможность создания связной мобильной ad hoc сети на основе смартфонов под управлением операционной системы iOS 10 и выше в условиях чрезвычайных ситуаций. Предполагается, что каждый такой смартфон оснащен передатчиками для работы с интерфейсами Wi-Fi, Bluetooth, GPS, Глонасс.

Для решения поставленной задачи организована прозрачная автоматическая система выбора альтернативных каналов связи с возможностью локализации соседних достижимых узлов для трансляции им своих координат. Предполагается, что узлы-соседи (как минимум один из них) имеют доступ к глобальной сети и могут сообщать имеющуюся информацию в соответствующие специальные (спасательные) службы.

В статье рассмотрен механизм взаимодействия узлов на основе фреймворка Multipeer Connectivity. С использованием этого фреймворка и программного модуля Bonjour, представляющего протокол автоматического обнаружения сервисов, разработана система информирования о чрезвычайных ситуациях. Система предоставляет возможность развертывания ad hoc сети без непосредственного выхода в глобальную сеть каждого ее участника, локализацию зоны ЧС, отправку сообщения SOS всем подключенным узлам. Если в текущей сессии не было достижимых узлов, то трансляция отодвигается по времени до тех пор, пока не произойдет подключение любого узла к данной сессии.

Ключевые слова: связь мобильных устройств, находящихся в непосредственной близости, Wi-Fi, Bluetooth, оппортунистические сети, самоорганизующиеся сети, Multipeer Connectivity, сессия, диаграмма классов.

Для цитирования: Приходько, Т. А. Система информирования о чрезвычайных ситуациях на основе ad hoc сетей / Т. А. Приходько, Н. А. Жудин, С. А. Воробьев. – DOI 10.25559/SITITO.16.202001.118-126 // Современные информационные технологии и ИТ-образование. – 2020. – Т. 16, № 1. – С. 118-126.

© Приходько Т.А., Жудин Н.А., Воробьев С.А., 2020



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Emergency Informing System Based on Ad hoc Networks

T. A. Prikhodko, N. A. Zhudin*, S. A. Vorobyov

Kuban State University, Krasnodar, Russia

149 Stavropolskaya St., Krasnodar 350040, Russia

* dantes04015380209@gmail.com

Abstract

One current trend for providing delay-tolerant communications under highly-dynamic routing conditions is to form ad hoc OppNets with the exploitation of smart mobile devices carried by people. Modern smartphones hide a great potential for emergency monitoring and management: they are truly pervasive, they can establish peer-to-peer wireless links using short-range communication technologies, thus guaranteeing coverage even when fixed infrastructures are unavailable, they can define the location with high accuracy through several embedded sensors and send messages without global connectivity.

The paper examines the possibility of creating a mobile ad hoc network based on smartphones running the iOS 10 operating system and higher in emergencies, providing that each smartphone is equipped with transmitters for working with Wi-Fi, Bluetooth, GPS, and GLONASS interfaces. To solve this problem, a transparent automatic system for selecting alternative communication channels is organized with the possibility of localizing neighboring reachable nodes to broadcast coordinates to them with further reporting the available information to the appropriate special (rescue) services.

The article describes the mechanism of nodes' interaction based on the Multipeer Connectivity framework. With this framework and the Bonjour software module, giving a protocol for automatic detection of services, an emergency informing system has been developed. This system provides the ability to deploy ad hoc network without directly accessing the global network of each of its participants, to localize the emergency zone, and to send an SOS- message to all connected peers. If there were no peers in the current session, the broadcast is pushed back in time until any peer is connected to this session.

Keywords: Mobile Networking in Proximity, Wi-Fi, Bluetooth, opportunistic networks, Ad-hoc networks, Multipeer Connectivity, Session, Class Diagram.

For citation: Prikhodko T.A., Zhudin N.A., Vorobyov S.A. Emergency Informing System based on Ad hoc Networks. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2020; 16(1):118-126. DOI: <https://doi.org/10.25559/SITITO.16.202001.118-126>



Введение

Появившиеся в прошлом десятилетии оппортунистические сети (OppNets) [1,2] обеспечили наличие устойчивой к задержкам связи в условиях высокодинамичной маршрутизации [3-6]. Одной из современных тенденций является формирование ad hoc [7,8] OppNets с использованием интеллектуальных мобильных устройств, в частности смартфонов. Однако, широко используемые мобильные операционные системы (OS), такие как Android и iOS, на сегодня все еще имеют существенные ограничения для организации беспроводных самоорганизующихся сетей [9,10]. В частности, общие интерфейсы WLAN/WPAN такие как Wi-Fi, Bluetooth и Bluetooth Low Energy (BLE) на этих OS не имеют достаточно возможностей для оппортунистической маршрутизации из-за постоянно меняющейся расположения устройств и плотности сети. В научных кругах существуют разработки ПО для поддержки связи между узлами ad hoc сети, как на основе OS Android [13-18], существуют также кроссплатформенные варианты ПО - Android и iOS¹ [10,11] – это Cocomoon и OpenGarden.

Таким образом, развёртывание беспроводной самоорганизующейся сети при чрезвычайных ситуациях является довольно сложной задачей, которую нужно решать в короткие сроки в режиме быстрого реагирования. Одной из главных и ключевых проблем является отсутствие функциональной совместимости между устройствами, работающими с помощью разных протоколов, а также имеющих разную мощность установленных передатчиков. Помимо этого может возникнуть ситуация, при которой ни одно из устройств сети не имеет доступа в глобальную сеть. Задача такой сети поддерживать максимально возможную связность при условии подвижности узлов и возможного отключения того или иного передатчика.

В работе мы исходим из предположения, что заблудившегося человека начинают искать поисковые группы, каждый участник которой снабжен современным устройством связи. В этих условиях на основе разработанного приложения сеть создается спонтанно, радиус действия каждого устройства охватывает зону действия протокола беспроводной передачи данных Bluetooth или Wi-Fi – от 100 до 300 метров.

Цели работы

В работе рассматривается возможность создания связной мобильной ad hoc сети на основе смартфонов под управлением операционной системы iOS 10 и выше в условиях чрезвычайных ситуаций. Предполагается, что каждый такой смартфон оснащен передатчиками для работы с интерфейсами Wi-Fi (802.11 a/b/g/n/ac), Bluetooth 4.0, GPS, Глонасс.

Для решения поставленной задачи необходимо организовать прозрачную, автоматическую систему выбора альтернативных каналов связи с возможностью локализации соседних достижимых узлов для трансляции им своих координат. Пред-

полагается, что узлы-соседи (как минимум один из них) имеют доступ к глобальной сети и могут сообщать имеющуюся информацию в соответствующие специальные (спасательные) службы. Подобная задача в некоторых источниках носит название Mobile Networking in Proximity (MNP)[12] и призвана обеспечить связь мобильных устройств, находящихся в непосредственной близости (на расстоянии от 100 до 300 метров).

Концепция Multipeer Connectivity

Одним из возможных средств для обеспечения оппортунистической связи мобильных устройств, находящихся в непосредственной близости, является фреймворк Multipeer Connectivity², который появился в операционной системе iOS 7, и его главной задачей стало обеспечение возможности установить сеанс связи в одноранговой сети по протоколам Wi-Fi и Bluetooth и работать со следующими логическими абстракциями: одноранговый узел (peer), сеанс, слушатель (Browser), маяк (Advertiser)^{3,4}. Диаграмма, отражающая взаимосвязь указанных сущностей показана на рис.1.

Сессия, которая инициализируется параметрами безопасности и шифрования, а также объектом класса MCPeerID, который имеет лишь одно свойство — displayName – это имя устройства, которое отображается для других соседних узлов. Его можно считать также названием ID-сессии. Чтобы текущую сессию могли видеть другие пользователи, фреймворк предоставляет класс MCAdvertiserAssistant, маяк, сообщающий всем об этой сессии и хранящий информацию о ней.

Как только слушатель захочет подключиться к сессии, маяк автоматически покажет уведомление об этом, с опциями «разрешить / отклонить подключение». Как только пользователь-Advertiser разрешит подключиться слушателю, тот попадет в сессию и получает возможность общения.

Сеанс (за него отвечает класс MCSession) касается только двух устройств. Третье устройство не может быть подключено к существующему сеансу.

Для слушателя разработчик с минимальными усилиями пишет код стороны слушателя, используя встроенный контроллер фреймворка MCBrowserViewController, который полностью берет на себя всю логику поиска и подключения к маяку или реализуют собственный контроллер.

Фреймворк является верхним уровнем над Bonjour – программным модулем, представляющим собой протокол автоматического обнаружения сервисов и служб. Bonjour позволяет реализовать с помощью протоколов стека TCP/IP объявление сервисов в локальных сетях для их легкого обнаружения, или обнаруживать другие, уже объявленные сервисы. Данный программный модуль реализован двумя классами NetService и NetServiceBrowser, а также интерфейсами NetServiceDelegate и NetServiceBrowserDelegate соответственно.

Класс NetService⁵ представляет сетевой сервис, который публикует приложение или является клиентом, т.е. позволяет

¹ Open Garden – Connecting the next billion mobile devices [Электронный ресурс]. URL: <https://opengarden.com> (дата обращения: 10.01.2020).

² MultipeerConnectivity Apple Developer [Электронный ресурс]. URL: <https://developer.apple.com/documentation/multipeerconnectivity> (дата обращения: 10.01.2020).

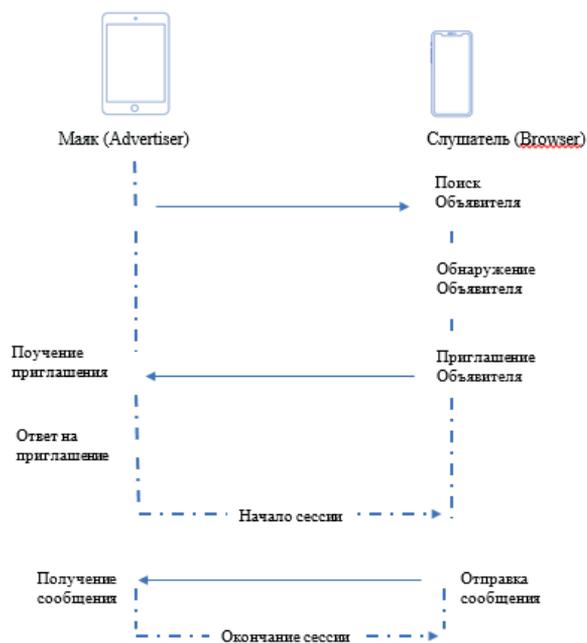
³ MCNearbyServiceBrowserDeveloper [Электронный ресурс]. URL: <https://developer.apple.com/documentation/multipeerconnectivity/mcnearbyservicebrowser> (дата обращения: 10.01.2020).

⁴ MCBrowserViewController [Электронный ресурс]. URL: <https://developer.apple.com/documentation/multipeerconnectivity/MCBrowserViewController> (дата обращения: 10.01.2020).

⁵ NetService Apple Developer [Электронный ресурс]. URL: <https://developer.apple.com/documentation/foundation/netservice> (дата обращения: 10.01.2020).



наш узел сделать видимым для соседних. Этот класс и класс NetServiceBrowser используют многоадресный DNS для передачи информации о сетевых службах в используемое приложение в виде узла сети. API NetService предоставляет удобный способ публикации сервисов.



Р и с. 1. Диаграмма работы устройств согласно Multipeer Connectivity
F i g. 1. Device operation diagram according to Multipeer Connectivity

HTTP и FTP – это две службы, которые обычно предоставляются системами. Приложения также могут определять свои собственные пользовательские службы для предоставления конкретных данных клиентам.

Класс NetService можно использовать как в качестве издателя службы, так и в качестве клиента службы. Если приложение публикует службу, код программы должен получить порт и подготовить сокет для связи с клиентами. Как только сокет готов, для уведомления об этом используется класс NetService. Для опубликования службы инициализируется объект NetService с именем службы, доменом, типом и информацией о порте. Вся эта информация должна быть действительной для сокета, созданного приложением. После инициализации вызывается метод publish(), выполняющий широковещательную передачу информации об услуге в сеть.

Методы NetService работают асинхронно, поэтому скорость работы сети не влияет на скорость работы приложения, что важно в случае, когда нужно быстро реагировать и при этом не блокировать интерфейс.

NetServiceBrowser⁶ используется для получения списка доступных доменов, а затем получения объект NetService для каждого обнаруженного узла (службы). Каждый браузер сетевой службы выполняет один поиск за раз, поэтому, если нужно выполнить несколько поисков одновременно, требуется несколько браузеров сетевых служб.

Браузер сетевой службы выполняет все поиски асинхронно, используя текущий цикл выполнения в фоновом режиме, пока объект не получит сообщение об окончании задачи.

Таким образом фреймворк MultipeerConnectivity, работающий с помощью Bonjour позволяет реализовать взаимодействие между узлами: динамически устанавливать соединение в среде передачи данных, быстро реагировать на смену соседних узлов и реализовать динамическую маршрутизацию в требуемой сети.

Логическая модель системы информирования о чрезвычайных ситуациях

Разработанная авторами система работает на платформе iOS версии 10 и выше. В ее составе использованы следующие библиотеки: MultipeerConnectivity, CoreLocation, GoogleMaps. Связывание между устройствами происходит на основе протоколов Wi-Fi или Bluetooth, то есть без доступа к глобальной сети Интернет.

Рассмотрим логическую модель разрабатываемой системы в виде приложения. В целом система представляет собой два приложения:

1. приложение для поисковых групп;
2. приложение для обозначения SOS-сигнала.

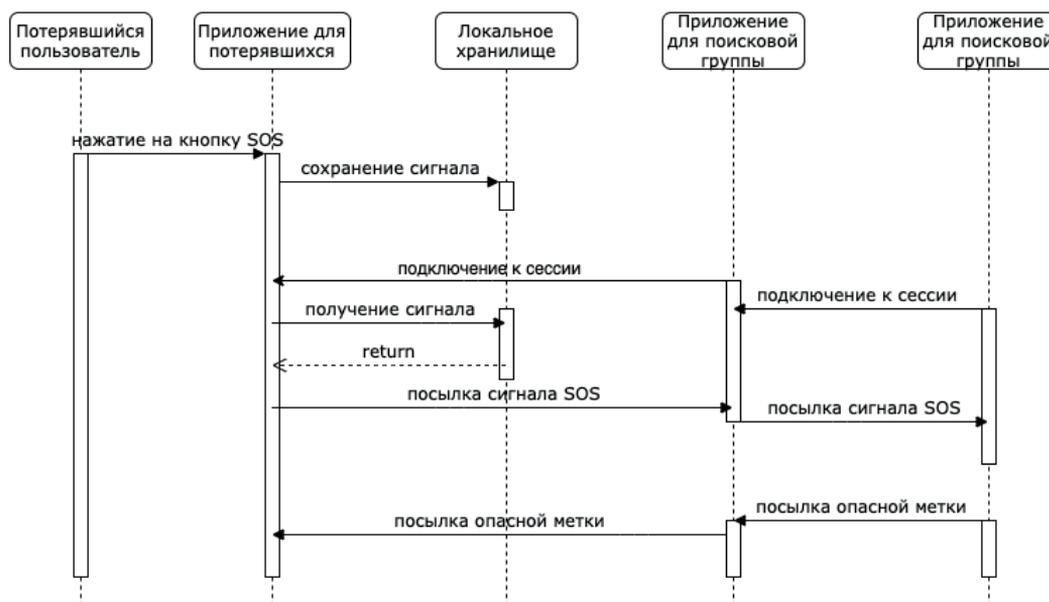
SOS-сигнал представляет собой точку на карте с координатами. В приложении есть специальная кнопка, по нажатию которой с помощью встроенной системной библиотеки CoreLocation, определяется текущая геопозиция пользователя. Устройство, с которого инициализируется подача SOS-сигнала, сохраняет этот сигнал в локальном хранилище и затем транслирует с помощью встроенной библиотеки MultipeerConnectivity всем подключенным узлам, а также новым по мере их присоединения к сессии (узлы появляются по мере приближения участников поисковой группы). При этом не требуется повторная инициализация SOS-сигнала, так как он хранится локально на устройстве. Последующие подключенные узлы также будут получать SOS-сигналы. Устройство, которое инициализирует SOS-сигнал, имеет локальное хранилище, содержащее уникальные имена устройств в рамках сессии, которые уже получили сигнал. Таким образом ликвидируется излишняя посылка данных и тем самым экономия разряда аккумулятора устройства (рис.2).

Другим маркером на карте является зона опасности (на рис.2 – «опасная метка»), которую могли заметить пользователи во время рейда по местности. Эти «опасные метки» имеют информацию о типе ЧС и сопровождаются кратким описанием и диаметром опасной зоны, которая представляет собой окружность. Эти маркеры, как и предыдущие транслируются на все устройства вне зависимости от типа приложения.

На все устройства с приложением поисковых групп происходит трансляция передвижения каждого устройства в локальной сети, чтобы на карте была актуальная разметка пройденных территорий. На устройства, использующие приложение для обозначения SOS-сигнала, трансляция данных координат передвижения не идет. Это сделано для экономии заряда устройства, которое в отличие от поисковых групп может не иметь дополнительных источников для зарядки устройства.

⁶ NetServiceBrowser Apple Developer [Электронный ресурс]. URL: <https://developer.apple.com/documentation/foundation/netservicebrowser> (дата обращения: 10.01.2020).





Р и с. 2. Диаграмма последовательности работы системы информирования о чрезвычайных ситуациях

F i g. 2. Emergency reporting system sequence diagram

Основные встроенные классы

Рассмотрим основные классы высокого уровня библиотеки MultipeerConnectivity, которые используются для реализации логической составляющей системы. Последовательность их работы отражена на рис.1.

Класс MCNearbyServiceAdvertiser публикует объявление для конкретного сервиса, которое приложение предоставляет через MultipeerConnectivity и уведомляет своего делегата о приглашениях от ближайших узлов. Сервис в данном случае представляет собой метаданные, которые описывают его предназначение (маяк на рис.1).

Класс MCNearbyServiceBrowser занимается поиском (по типу) сервисов, предлагаемых близлежащими устройствами, использующими инфраструктуру Wi-Fi, peer-to-peer Wi-Fi и Bluetooth (в iOS) или Ethernet (в MacOS и tvOS), и предоставляет возможность легко пригласить эти устройства на сеанс MultipeerConnectivity (MCSession) (слушатель на рис.1).

MCSession – класс, который разрешает и управляет коммуникацией между всеми участниками в сессии MultipeerConnectivity. С помощью следующей функции send(_ data: Data, toPeers peerIDs: [MCPeerID], with mode: MCSessionSendDataMode) происходит отправка данных массиву узлов (устройства, располагающиеся в одной сессии), представляющих собой уникальные имена участников. Используя свойство объектов этого класса connectedPeers, может быть получен весь список подключенных участников. Получение сообщения можно зафиксировать используя функцию делегата MCSession – session(_ session: MCSession, didReceive data: Data, fromPeer peerID: MCPeerID) (тело сессии на рис.1).

У библиотеки MultipeerConnectivity есть небольшой минус – если устройства связаны между собой не напрямую, а через несколько узлов, то посылка сообщений не вызывает функ-

цию приема сообщения у другого устройства⁷. Для решения этой проблемы была реализована своя система высокого уровня для возможности ретрансляции данных к устройствам, находящимся на уровне выше одного хопа. Эта система достаточно проста. При вызове функции приема сообщения, устройство инициализирует отправку этого сообщения своим участникам сессии. Выборка участников делается следующим образом.

В работе приняты следующие обозначения:

$Peers_i$ – множество узлов, подключенных к устройству i ; $Sending_i$ – множество узлов, которые инициализировали отправку данных в рамках одной транзакции всей ретрансляции i -му устройству (хранится в теле сообщения и каждый раз модернизируется устройством при ретрансляции, добавляя себя в это множество). Тогда для устройства i – множество узлов, которым нужно ретранслировать данные = $Peers_i - Sending_i$.

Основные классы мобильного приложения системы

Мобильное приложение состоит из 7 основных классов. Диаграмма представлена на рисунке 3. Класс View отвечает за отображение графического интерфейса, его своевременное обновление, а также предоставляет контейнер для MapService, который нужен для инициализации карты. View имеет свой жизненный цикл, обработчики событий, которые передают сигнал в Presenter, который решает, как их обрабатывать. Presenter в свою очередь вызывает методы View для обновления графического интерфейса. Presenter использует помимо View различные сервисы:

LocationService – предоставляет информацию о текущей геолокации, следит за ее настройками.

⁷ Страх и ненависть в Multipeer Connectivity [Электронный ресурс]. URL: <https://habr.com/ru/company/dataart/blog/275627/> (дата обращения: 10.01.2020).

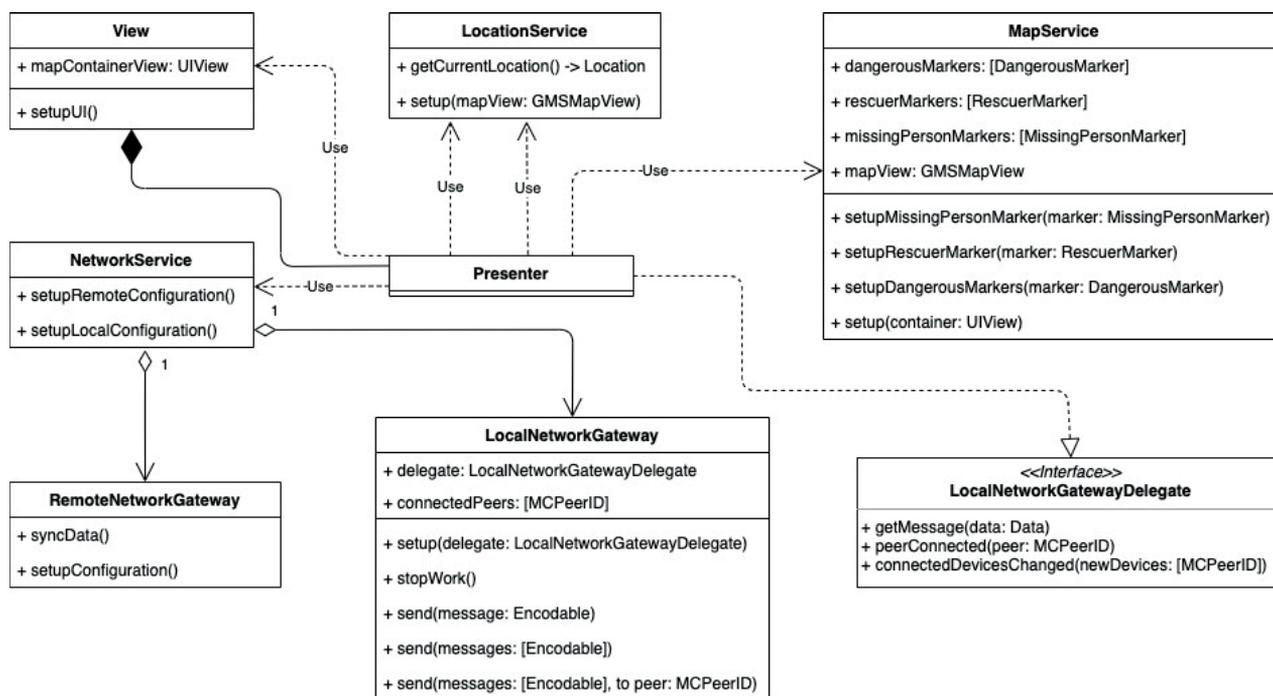


MapService – класс, предоставляющий в удобном виде API для управлением картой, а именно: настройка маркеров (точки на карте с необходимой информацией) различного типа. Также этот класс хранит все маркеры.

NetworkService – инкапсулирует в себе работу двух типов соединения: с выходом в глобальную сеть (RemoteNetworkGateway) и в рамках локальной сети (LocalNetworkGateway). Класс также

занимается настройкой конфигураций используемых инструментов связи в RemoteNetworkGateway и LocalNetworkGateway. Также проксирует API из этих gateway.

RemoteNetworkGateway – позволяет приложению выйти в глобальную сеть и синхронизировать данные. Имеет под собой встроенный класс URLSession. В рамках данной работы эта абстракция не реализована.



Р и с. 3. Диаграмма классов мобильного приложения
F i g. 3. Mobile application class diagram

LocalNetworkGateway – класс, позволяющий взаимодействовать с другими участниками локальной сети с помощью встроенного MultipeerConnectivity. Здесь происходит настройка параметров и обработка поступающих событий: принятие сообщения или изменение состояния сети.

Обработку всех событий от LocalNetworkGateway принимает на себя Presenter, реализуя интерфейс LocalNetworkGatewayDelegate.

Описание интерфейса приложения

Рассмотрим прототипы интерфейсов приложений. На рисунке 3а изображена карта, на которой показаны все маркеры. На данном экране посредством прикосновения можно оставить предупреждение об опасности, и после заполнения информации, подтвердив кнопкой «Сообщить». На рисунке 3б показан экран приложения пропавшего человека, который может инициировать SOS-сигнал.



а) карта маркеров, б) генерация SOS-сигнала
Р и с. 3. Интерфейс приложения:
a) marker map, б) SOS signal generation



Варианты развития приложения

Предполагается разработка серверной части приложения, которая будет выполнять роль глобального хранилища различного рода данных:

1. месторасположения чрезвычайных ситуаций с необходимым минимальным описанием;
2. месторасположения поисковых групп (от начала и до конца следования);
3. месторасположения SOS-сигналов.

Эти данные, хранящиеся на выделенном сервере, можно использовать для трансляции различным службам, которые занимаются ликвидацией чрезвычайных ситуаций, а также волонтерским организациям, которые выделяют ресурсы на поиски пропавших людей.

Сервер планируется связать со всеми мобильными приложениями, которые смогли получить доступ к Интернету. Также в мобильном приложении, которое смогло получить доступ к Интернету на развернутой местности, будет происходить автоматическое обновление данных о всех месторасположениях в заданной области, которая настраивается в приложении. Предполагается очерчивание окружности с заданным радиусом и координатами центра. Автоматическое обновление данных с сервера на одном устройстве повлечет за собой глобальное обновление всех устройств в локальной сети.

Предлагается также создание чата между участниками в рамках локальной сети.

Заключение

В результате работы была разработана система, которая предоставляет возможность развертывания беспроводной сети в режиме ad hoc при чрезвычайных ситуациях без непосредственного выхода в глобальную сеть каждого ее участника.

Были проанализированы особенности взаимодействия между различными устройствами в рамках локальной сети. Выявлено слабое место MultipeerConnectivity, подтолкнувшее к реализации возможности передачи сообщения через промежуточного участника сети. На выходе было получено два прикладных приложения, которые предоставляют одну из функциональностей, необходимых в условиях ЧС, а именно: обмен координатами передвижения, маркерами опасности и SOS-сигналами.

Список использованных источников

- [1] Pelusi, L. Opportunistic networking: data forwarding in disconnected mobile ad hoc networks / L. Pelusi, A. Passarella, M. Conti. – DOI 10.1109/MCOM.2006.248176 // IEEE Communications Magazine. – 2006. – Vol. 44, No. 11. – Pp. 134-141. – URL: <https://ieeexplore.ieee.org/document/4014485> (дата обращения: 10.01.2020).
- [2] Guo, B. Opportunistic IoT: Exploring the social side of the internet of things / B. Guo, Z. Yu, X. Zhou, D. Zhang. – DOI 10.1109/CSCWD.2012.6221932 // Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD). – Wuhan, 2012. – Pp. 925-929. – URL: <https://ieeexplore.ieee.org/document/6221932> (дата обращения: 10.01.2020).
- [3] Григорьев, А. А. Алгоритмы маршрутизации в оппортунистических сетях / А. А. Григорьев // Труды московского физико-технического института. – 2013. – Т. 5, № 3. – URL: <https://www.elibrary.ru/item.asp?id=19416894> (дата обращения: 10.01.2020). – Рез. англ.
- [4] Govindasamy, J. A comparative study of reactive, proactive and hybrid routing protocol in wireless sensor network under wormhole attack / J. Govindasamy, S. Punniakody. – DOI 10.1016/j.jesit.2017.02.002 // Journal of Electrical Systems and Information Technology. – 2018. – Vol. 5, issue 3. – Pp. 735-744. – URL: <https://www.sciencedirect.com/science/article/pii/S2314717217300181> (дата обращения: 10.01.2020).
- [5] Chitkara, M. Review on MANET: Characteristics, Challenges, Imperatives and Routing Protocols / M. Chitkara, M. E. Ahmad // International Journal of Computer Science and Mobile Computing. – 2014. – Vol. 3, issue 2. – Pp. 432-437. – URL: <https://ijcsmc.com/docs/papers/February2014/V3I2201499a3.pdf> (дата обращения: 10.01.2020).
- [6] Kaur, E. J. Review study on MANET Routing Protocols: Challenges and Applications / E. J. Kaur, E. G. Singh. – DOI 10.26483/ijarcs.v8i4.3718 // International Journal of Advanced Research in Computer Science. – 2017. – Vol. 8, issue 4. – Pp. 140-145. – URL: <http://www.ijarcs.info/index.php/ijarcs/article/view/3718> pdf (дата обращения: 10.01.2020).
- [7] Гусс, С. В. Самоорганизующиеся mesh-сети для частного использования / С. В. Гусс // Математические структуры и моделирование. – 2016. – № 4(40). – С. 102-115. – URL: <https://www.elibrary.ru/item.asp?id=27405154> (дата обращения: 10.01.2020). – Рез. англ.
- [8] Смирнова, Е. В., и др. Технологии современных беспроводных сетей Wi-Fi / Е. В. Смирнова, А. В. Пролетарский и др.; под общ. ред. А. В. Пролетарского. – Москва: Изд-во МГТУ им. Н.Э. Баумана, 2017. – 446 с.
- [9] Nishiyama, H. Relay-by-smartphone: realizing multihop device-to-device communications / H. Nishiyama, M. Ito, N. Kato. – DOI 10.1109/MCOM.2014.6807947 // IEEE Communications Magazine. – 2014. – Vol. 52, issue 4. – Pp. 56-65. – URL: <https://ieeexplore.ieee.org/document/6807947> (дата обращения: 10.01.2020).
- [10] Turkes, O. Cocoon: A lightweight opportunistic networking middleware for community-oriented smart mobile applications / O. Turkes, H. Scholten, P.J.M. Havinga. – DOI 10.1016/j.comnet.2016.08.021 // Computer Networks. – 2016. – Vol. 111. – Pp. 93-108. – URL: <https://www.sciencedirect.com/science/article/pii/S1389128616302742> (дата обращения: 10.01.2020).
- [11] Aloï, G. The SENSE-ME platform: Infrastructure-less smartphone connectivity and decentralized sensing for emergency management / G. Aloï, O. Briante, M. D. Felice, G. Ruggeri, S. Savazzi. – DOI 10.1016/j.pmcj.2017.10.004 // Pervasive and Mobile Computing. – 2017. – Vol. 42. – Pp. 187-208. – URL: <https://www.sciencedirect.com/science/article/pii/S1574119217300214> (дата обращения: 10.01.2020).
- [12] Holzer, A. Padoc: Enabling social networking in proximity / A. Holzer, S. Reber, J. Quarta, J. Mazuze, D. Gillet. – DOI 10.1016/j.comnet.2016.08.009 // Computer Networks. – 2016. – Vol. 111. – Pp. 82-92. – URL: <https://www.sciencedirect.com/science/article/pii/S138912861630250X>



- (дата обращения: 10.01.2020).
- [13] Gardner-Stephen, P. MeshMS: Ad Hoc Data Transfer with in Mesh Network / P. Gardner-Stephen, J. Lakeman, R. Challans, C. Wallis, A. Stulman, Y. Haddad. – DOI 10.4236/ijcns.2012.58060 // International Journal of Communications, Network and System Sciences. – 2012. – Vol. 5, issue 8. – Pp. 496-504. – URL: <https://www.scirp.org/journal/paperinformation.aspx?paperid=21934> (дата обращения: 10.01.2020).
- [14] Gardner-Stephen, P. Serval mesh software-WiFi multi model management / P. Gardner-Stephen, S. Palaniswamy. – DOI 10.1145/2185216.2185245 // Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief (ACWR'11). – Association for Computing Machinery, New York, NY, USA, 2011. – Pp. 71-77. – URL: <https://dl.acm.org/doi/10.1145/2185216.2185245> (дата обращения: 10.01.2020).
- [15] Boldrini, C. Context- and social-aware middleware for opportunistic networks / C. Boldrini, M. Conti, F. Delmastro, A. Passarella. – DOI 10.1016/j.jnca.2010.03.017 // Journal of Network and Computer Applications. – 2010. – Vol. 33, issue 5. – Pp. 525-541. – URL: <https://www.sciencedirect.com/science/article/pii/S1084804510000524> (дата обращения: 10.01.2020).
- [16] Trifunovic, S. WLAN-Opp: Ad-hoc-less opportunistic networking on smartphones / S. Trifunovic, M. Kurant, K. A. Hummel, F. Legendre. – DOI 10.1016/j.adhoc.2014.07.011 // Ad Hoc Networks. – 2015. – Vol. 25, Part B. – Pp. 346-358. – URL: <https://www.sciencedirect.com/science/article/pii/S1570870514001425> (дата обращения: 10.01.2020).
- [17] Al-Akkad, A. Help beacons: design and evaluation of an ad-hoc lightweight s.o.s. system for smartphones / A. Al-Akkad, I. Ramirez, A. Boden, D. Randall, A. Zimmermann. – DOI 10.1145/2556288.2557002 // Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14). – Association for Computing Machinery, New York, NY, USA, 2014. – Pp. 1485-1494. – URL: <https://dl.acm.org/doi/10.1145/2556288.2557002> (дата обращения: 10.01.2020).
- [18] Danieleto M., Quer G., Rao R.R., Zorzi M. On the Exploitation of the Android OS for the Design of a Wireless Mesh Network Testbed / M. Danieleto, G. Quer, R. R. Rao, M. Zorzi. – DOI 10.1109/MILCOM.2013.179 // MILCOM 2013 - 2013 IEEE Military Communications Conference. – San Diego, CA, 2013. – Pp. 1032-1038. – URL: <https://ieeexplore.ieee.org/document/6735760> (дата обращения: 10.01.2020).
- [19] Allen, S. M. Exploiting user interest similarity and social links for micro-blog forwarding in mobile opportunistic networks / S. M. Allen, M. J. Chorley, G. B. Colombo, E. Jaho, M. Karaliopoulos, I. Stavrakakis, R. M. Whitaker. – DOI 10.1016/j.pnmcj.2011.12.003 // Pervasive and Mobile Computing. – 2014. – Vol. 11. – Pp. 106-131. – URL: <https://www.sciencedirect.com/science/article/pii/S1574119211001544> (дата обращения: 10.01.2020).
- [20] Fok, C.-L. Servilla: A flexible service provisioning middleware for heterogeneous sensor networks / C.-L. Fok, G.-C. Roman, C. Lu. – DOI 10.1016/j.scico.2010.11.006 // Science of Computer Programming. – 2012. – Vol. 77, issue 6. – Pp. 663-684. – URL: <https://www.sciencedirect.com/science/article/pii/S0167642310002054> (дата обращения: 10.01.2020).
- ence/article/pii/S0167642310002054 (дата обращения: 10.01.2020).
- Поступила 10.01.2020; принята к публикации 20.03.2020; опубликована онлайн 25.05.2020.*

Об авторах:

Приходько Татьяна Александровна, доцент кафедры вычислительных технологий, факультет компьютерных технологий и прикладной математики, Кубанский государственный университет (350040, Россия, г. Краснодар, ул. Ставропольская, д. 149), кандидат технических наук, ORCID: <http://orcid.org/0000-0001-5137-2064>, pr.tatyana@gmail.com

Жудин Никита Андреевич, магистрант кафедры вычислительных технологий, факультет компьютерных технологий и прикладной математики, Кубанский государственный университет (350040, Россия, г. Краснодар, ул. Ставропольская, д. 149), ORCID: <http://orcid.org/0000-0002-9299-4051>, dantes04015380209@gmail.com

Воробьев Семен Анатольевич, магистрант кафедры вычислительных технологий, факультет компьютерных технологий и прикладной математики, Кубанский государственный университет (350040, Россия, г. Краснодар, ул. Ставропольская, д. 149), ORCID: <http://orcid.org/0000-0001-8925-4709>, babaev.danil.vs@gmail.com

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

- [1] Pelusi L., Passarella A., Conti M. Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. *IEEE Communications Magazine*. 2006; 44(11):134-141. (In Eng.) DOI: <https://doi.org/10.1109/MCOM.2006.248176>
- [2] Guo B., Yu Z., Zhou X., Zhang D. Opportunistic IoT: Exploring the social side of the internet of things. In: *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, Wuhan; 2012. p. 925-929. (In Eng.) DOI: <https://doi.org/10.1109/CSCWD.2012.6221932>
- [3] Grigoriev A.A. Opportunistic Routing Algorithms. *Proceedings of MIPT*. 2013; 5(3):112-120. Available at: <https://elibrary.ru/item.asp?id=19416894> (accessed 10.01.2020). (In Russ., abstract in Eng.)
- [4] Govindasamy J., Punniakody S. A comparative study of reactive, proactive and hybrid routing protocol in wireless sensor network under wormhole attack. *Journal of Electrical Systems and Information Technology*. 2018; 5(3):735-744. (In Eng.) DOI: <https://doi.org/10.1016/j.jesit.2017.02.002>
- [5] Chitkara M., Ahmad M.E. Review on MANET: Characteristics, Challenges, Imperatives and Routing Protocols. *International Journal of Computer Science and Mobile Computing*. 2014; 3(2):432-437. Available at: <https://ijcsmc.com/docs/papers/February2014/V3I2201499a3.pdf> (accessed 10.01.2020). (In Eng.)
- [6] Kaur E.J., Singh E.G. Review study on MANET Routing Protocols: Challenges and Applications. *International Journal of Advanced Research in Computer Science*. 2017; 8(4):140-



145. (In Eng.) DOI: <https://doi.org/10.26483/ijarcs.v8i4.3718>
- [7] Guss S.V. Private Wireless Mesh Networks. *Mathematical Structures and Modeling*. 2016; (4):102-115. Available at: <https://www.elibrary.ru/item.asp?id=27405154> (accessed 10.01.2020). (In Russ., abstract in Eng.)
- [8] Smirnova E.V., Proletarsky A.V. et al. *Tehnologii sovremennyh besprovodnyh setej Wi-Fi* [Technologies of modern wireless Wi-Fi networks]. Moscow, BMSTU Publ.; 2017. (In Russ.)
- [9] Nishiyama H., Ito M., Kato N. Relay-by-smartphone: realizing multihop device-to-device communications. *IEEE Communications Magazine*. 2014; 52(4):56-65. (In Eng.) DOI: <https://doi.org/10.1109/MCOM.2014.6807947>
- [10] Turkes O., Scholten H., Havinga P.J.M. Cocoon: A lightweight opportunistic networking middleware for community-oriented smart mobile applications. *Computer Networks*. 2016; 111:93-108. (In Eng.) DOI <https://doi.org/10.1016/j.comnet.2016.08.021>
- [11] Aloï G., Briante O., Felice M.D., Ruggeri G., Savazzi S. The SENSE-ME platform: Infrastructure-less smartphone connectivity and decentralized sensing for emergency management. *Pervasive and Mobile Computing*. 2017; 42:187-208. (In Eng.) DOI: <https://doi.org/10.1016/j.pmcj.2017.10.004>
- [12] Holzer A., Reber S., Quarta J., Mazuze J., Gillet D. Padoc: Enabling social networking in proximity. *Computer Networks*. 2016; 111:82-92. (In Eng.) DOI: <https://doi.org/10.1016/j.comnet.2016.08.009>
- [13] Gardner-Stephen P., Lakeman J., Challans R., Wallis C., Stulman A., Haddad Y. MeshMS: Ad Hoc Data Transfer within Mesh Network. *International Journal of Communications, Network and System Sciences*. 2012; 5(8):496-504. (In Eng.) DOI: <https://doi.org/10.4236/ijcns.2012.58060>
- [14] Gardner-Stephen P., Palaniswamy S. Serval mesh software-WiFi multi model management. In: *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief (ACWR'11)*. Association for Computing Machinery, New York, NY, USA; 2011. p. 71-77. (In Eng.) DOI: <https://doi.org/10.1145/2185216.2185245>
- [15] Boldrini C., Conti M., Delmastro F., Passarella A. Context- and social-aware middleware for opportunistic networks. *Journal of Network and Computer Applications*. 2010; 33(5):525-541. (In Eng.) DOI: <https://doi.org/10.1016/j.jnca.2010.03.017>
- [16] Trifunovic S., Kurant M., Hummel K.A., Legendre F. WLAN- Opp: Ad-hoc-less opportunistic networking on smartphones. *Ad Hoc Networks*. 2015; 25-B:346-358. (In Eng.) DOI: <https://doi.org/10.1016/j.adhoc.2014.07.011>
- [17] Al-Akkad A., Ramirez L., Boden A., Randall D., Zimmermann A. Help beacons: design and evaluation of an ad-hoc lightweight s.o.s. system for smartphones. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA; 2014. p. 1485-1494. (In Eng.) DOI: <https://doi.org/10.1145/2556288.2557002>
- [18] Danieletto M., Quer G., Rao R.R., Zorzi M. On the Exploitation of the Android OS for the Design of a Wireless Mesh Network Testbed. In: *MILCOM 2013 - 2013 IEEE Military Communications Conference*, San Diego, CA; 2013. p. 1032-1038. (In Eng.) DOI: <https://doi.org/10.1109/MILCOM.2013.179>
- [19] Allen S.M., Chorley M.J., Colombo G.B., Jaho E., Karaliopoulou M., Stavarakakis I., Whitaker R.M. Exploiting user interest similarity and social links for micro-blog forwarding in mobile opportunistic networks. *Pervasive and Mobile Computing*. 2014; 11:106-131. (In Eng.) DOI: <https://doi.org/10.1016/j.pmcj.2011.12.003>
- [20] Fok C.-L., Roman G.-C., Lu C. Servilla: A flexible service provisioning middleware for heterogeneous sensor networks. *Science of Computer Programming*. 2012; 77(6):663-684. (In Eng.) DOI: <https://doi.org/10.1016/j.scico.2010.11.006>

Submitted 10.11.2020; revised 20.03.2020;
published online 25.05.2020.

About the authors:

Tatyana A. Prikhodko, Associate Professor of the Department of Computing Technologies, Faculty of Computer Technologies and Applied Mathematics, Kuban State University (149 Stavropolskaya St., Krasnodar 350040, Russia), Ph.D. (Technology), ORCID: <http://orcid.org/0000-0001-5137-2064>, pr.tatyana@gmail.com

Nikita A. Zhudin, Undergraduate student of the Department of Computing Technologies, Faculty of Computer Technologies and Applied Mathematics, Kuban State University (149 Stavropolskaya St., Krasnodar 350040, Russia), ORCID: <http://orcid.org/0000-0002-9299-4051>, dantes04015380209@gmail.com

Semyon A. Vorobyov, Undergraduate student of the Department of Computing Technologies, Faculty of Computer Technologies and Applied Mathematics, Kuban State University (149 Stavropolskaya St., Krasnodar 350040, Russia), ORCID: <http://orcid.org/0000-0001-8925-4709>, babaev.danil.vs@gmail.com

All authors have read and approved the final manuscript.

