



## Мобильное приложение для поиска оптимального маршрута в университетском городке

© 2023, С.Т. Дусакаева ✉, В.В. Савинов

Оренбургский государственный университет, Оренбург, Россия

### Аннотация

Рассмотрена задача разработки мобильного приложения для поиска оптимального маршрута в университетском городке. Математическую основу проектирования мобильного приложения составляет задача поиска кратчайшего пути в графе, вершинами которого являются учебные аудитории, лаборатории, библиотеки, кафедры, деканаты и т.п. в корпусах университета. Проведён анализ основных характеристик существующих на рынке мобильных приложений для пешей навигации, выявлены их достоинства и недостатки. Рассмотрены известные алгоритмы поиска кратчайшего пути в графе: алгоритм поиска в ширину, жадный алгоритм поиска кратчайшего пути, алгоритм Дейкстры. Для рассматриваемой задачи разработаны программная реализация и вспомогательный сервис по созданию карт и приведения их к формату, в котором автоматически создаётся граф для работы приложения. Новизна предложенной разработки заключается в возможности отображения, добавления и редактирования различных карт. Дальнейшее совершенствование разработанного мобильного приложения может быть осуществлено в направлении повышения точности навигации за счёт объединения алгоритмов поиска путей в графе, а также в переводе карт из 2D- в 3D-пространство. Результаты исследования могут быть использованы для поиска оптимальных маршрутов в других университетах.

**Ключевые слова:** навигация по картам, мобильное приложение, оптимальный маршрут, алгоритм, поиск кратчайшего пути в графе, университетский городок.

**Цитирование:** Дусакаева С.Т., Савинов В.В. Мобильное приложение для поиска оптимального маршрута в университетском городке // Онтология проектирования. 2023. Т.13, №3(49). С.455-464. DOI:10.18287/2223-9537-2023-13-3-455-464.

**Конфликт интересов:** авторы заявляют об отсутствии конфликта интересов.

### Введение

Крупные университеты имеют сложную архитектурную планировку, поэтому для обучающихся и гостей университета задача поиска нужной аудитории является актуальной. Например, университетский городок Оренбургского государственного университета (ОГУ) состоит из 14 учебных корпусов и сотен различных аудиторий [1].

В настоящее время активно используются навигационные сервисы в открытом пространстве во многих сферах деятельности. Всё большую актуальность приобретает навигация, основанная на использовании мобильных приложений (МП), что обусловлено доступностью мобильных устройств [2]. В контексте рассматриваемой проблемы наиболее известны следующие МП для пешей навигации: «Яндекс Карты» [3], 2ГИС [4], *Google Maps* [5], *HERE WeGo* [6], *Maps.me* [7], *Sygie GPS Navigation* [8], *Magic Earth Navigation & Maps* [9], *iGO Navigation* [10], *OsmAnd* [11], *MapFactor Navigation* [12].

Результаты проведённого анализа основных характеристик названных МП в виде выявленных достоинств для данной задачи приведены в таблице 1. Среди недостатков некоторых из рассмотренных МП можно отметить: притормаживающий пользовательский интерфейс; устаревшая графика; наличие рекламы; не возможность поиска оптимального пути и др. По

информации из открытых источников удалось лишь установить, что в МП [3] для построения маршрутов используется алгоритм Дейкстры.

Таблица 1 – Основные характеристики и достоинства МП

Мобильное приложение название, разработчик, сайт, платформа	Достоинства
<b>Яндекс. Карты</b> Яндекс mobile.yandex.ru/apps/maps Android; iOS	<ul style="list-style-type: none"> <li>▪ позволяет строить маршруты для пешеходов;</li> <li>▪ знает короткие пути через дворы и проходы между зданиями;</li> <li>▪ показывает входы в здания и включает поэтажные схемы;</li> <li>▪ имеет голосовой помощник «Яндекс. Алиса».</li> </ul>
<b>2ГИС</b> ДубльГИС info.2gis.com Android; iOS; Wear OS	<ul style="list-style-type: none"> <li>▪ умеет прокладывать самые короткие пешие маршруты;</li> <li>▪ может работать без подключения к Интернету;</li> <li>▪ отображает поэтажные схемы торговых центров (ТЦ), вокзалов и аэропортов;</li> <li>▪ содержит путеводитель с информацией о городских достопримечательностях и интересных местах.</li> </ul>
<b>Google Maps</b> Google google.com/maps Android; iOS	<ul style="list-style-type: none"> <li>▪ позволяет строить пешие маршруты и знает расписание общественного транспорта;</li> <li>▪ поддерживает технологию дополненной реальности (можно видеть подсказки поверх изображения окружающей местности);</li> <li>▪ предоставляет возможность просмотра панорамных снимков и схем, которые помогают ориентироваться в больших зданиях (аэропорты, вокзалы, ТЦ и др.);</li> <li>▪ поддерживает голосовое управление и сохранение карт в памяти устройства с целью последующего использования без подключения к Интернету.</li> </ul>
<b>HERE WeGo</b> Here Technologies here.com Android; iOS	<ul style="list-style-type: none"> <li>▪ за считанные секунды подбирает доступные варианты перемещения из начальной точки в конечную, рассчитывает примерное время в пути и позволяет выбрать оптимальный маршрут из списка;</li> <li>▪ при навигации может переключаться в 3D-режим;</li> <li>▪ имеется возможность сохранения карт в памяти мобильного устройства и использования приложения в автономном режиме.</li> </ul>
<b>Maps.me</b> Maps.me Limited maps.me Android; iOS	<ul style="list-style-type: none"> <li>▪ построен на основе свободной географической карты OpenStreetMap;</li> <li>▪ картографические данные можно скопировать в память мобильного устройства и пользоваться приложением без подключения к Интернету;</li> <li>▪ умеет строить пешие и велосипедные маршруты.</li> </ul>
<b>Sygy GPS Navigation</b> Sygy sygy.com/gps-navigation Android; iOS	<ul style="list-style-type: none"> <li>▪ поддерживается поиск по организациям.</li> </ul>
<b>Magic Earth Navigation &amp; Maps</b> General Magic magicearth.com Android; iOS	<ul style="list-style-type: none"> <li>▪ построен на основе свободной географической карты OpenStreetMap;</li> <li>▪ предусмотрен выбор альтернативных маршрутов и справочник организаций;</li> <li>▪ присутствует функция «Что находится рядом», которая выводит список близлежащих объектов.</li> </ul>
<b>iGO Navigation</b> NNG Software Developing and Commercial igonavigation.com Android; iOS	<ul style="list-style-type: none"> <li>▪ позволяет строить маршруты для пешеходов и автомобилистов.</li> </ul>
<b>OsmAnd</b> OsmAnd osmand.net Android; iOS	<ul style="list-style-type: none"> <li>▪ построен на основе свободной географической карты OpenStreetMap;</li> <li>▪ предоставляет возможность получения ежечасных обновлений карт;</li> <li>▪ имеет встроенный путеводитель;</li> <li>▪ ведет запись маршрута.</li> </ul>

Все рассмотренные МП преимущественно ориентированы для навигации на открытой местности большой площади. При этом внутреннюю навигацию в них предполагается ис-

пользовать для известных крупных сооружений (большие ТЦ, вокзалы, государственные учреждения и др.) с большими потоками людей. Учебные корпуса университетов, как правило, локализованы на территории небольшой площади. Это означает, что многие функции, заложенные в указанных навигационных сервисах, не будут использованы, а возможность добавления карт требуемых зданий может быть сопряжена с угрозой ослабления безопасности университета. При этом требуется обеспечить большую точность для построения маршрутов в случаях, когда исходная и искомая точки маршрута находятся в разных корпусах или на разных этажах. Важное значение может иметь отыскание оптимального маршрута внутри здания без выхода на улицу, например, в зимнее время года. Поэтому для поиска оптимального маршрута в университете требуется навигационный сервис, учитывающий перечисленные факторы.

Целью исследования является разработка МП для поиска оптимального маршрута в университетском городке.

Для достижения поставленной цели необходимо:

- сформулировать математическую задачу в виде поиска кратчайшего пути в графе;
- выбрать алгоритм поиска кратчайшего пути в графе;
- разработать логическую архитектуру программы МП;
- разработать вспомогательный сервис для добавления новых карт в МП;
- проверить работоспособность МП.

## 1 Постановка задачи и метод решения

Задачу поиска оптимального пути от исходной точки  $u$  до искомой аудитории  $v$  можно представить как известную задачу поиска кратчайшего пути в графе [13, 14]. Такой подход используется в рассмотренных МП [3-12].

Пусть задан ортограф  $G(V, A)$ , каждой дуге которого  $(u, v)$  ставится в соответствие число  $L(u, v)$  – длина дуги. Под длиной дуги маршрута (пути) понимают сумму дуг, составляющих маршрут. Требуется найти длины кратчайших маршрутов и маршруты от выбранной вершины  $u$  до всех остальных вершин графа  $v_i$ .

Наиболее известными алгоритмами поиска кратчайшего маршрута в графе являются: алгоритм поиска в ширину, жадный алгоритм поиска кратчайшего пути, алгоритм Дейкстры [15, 16]. В таблице 2 приведены результаты сравнительного анализа указанных алгоритмов.

Таблица 2 – Результаты сравнительного анализа алгоритмов отыскания кратчайшего пути в графе

Алгоритм	Достоинства	Недостатки
Алгоритм поиска в ширину	<ul style="list-style-type: none"> <li>▪ сравнительная простота реализации</li> <li>▪ полнота и оптимальность</li> </ul>	<ul style="list-style-type: none"> <li>▪ при увеличении пространства поиска производительность поиска снижается по сравнению с другими эвристическими алгоритмами</li> </ul>
Жадный алгоритм поиска	<ul style="list-style-type: none"> <li>▪ простота поиска оптимального решения</li> </ul>	<ul style="list-style-type: none"> <li>▪ сложность оценки алгоритма</li> </ul>
Алгоритм Дейкстры	<ul style="list-style-type: none"> <li>▪ сравнительная простота реализации</li> </ul>	<ul style="list-style-type: none"> <li>▪ находит кратчайшие пути и их величины только от одной вершины</li> </ul>

В качестве основы для построения математической модели задачи принят алгоритм Дейкстры. Программная реализация построена на основе *UML*-шаблонов проектирования [17, 18]. Архитектура программного средства *BLoC* (компонент бизнес-логики, от англ. *Business Logic Component*), идея которой заключается в разделении интерфейса (*UI*-компонента, от англ. *User Interface*) и бизнес-логики приложения, соответствует единой системы программной документации [19].

Для взаимодействия с пользователем требуются следующие визуальные элементы управления:

- поле выбора начальной и конечной точек маршрута;
- кнопки старта и сброса маршрута;
- кнопки для пролистывания этапов следования по маршруту;
- кнопки, отвечающие за масштабирование и ориентацию карты;
- поле для отображения карты и рассчитанного маршрута;
- выпадающий список для выбора отображаемой карты.

Для работы интерфейса требуются классы, в которых реализованы алгоритмы: поиска путей, взаимодействия с базой данных (БД), отображения карт и дополнительной информации, работы с временными данными. Термин «класс» в данном случае используется в контексте объектно-ориентированного программирования и подразумевает собой шаблон для создания объектов, обеспечивающий начальные значения состояний: инициализация полей-переменных и реализация поведения функций или методов. Это позволяет создать универсальный программный продукт для различных карт.

На рисунке 1 представлена вкладка «Навигация» МП. Первый *UI*-компонент (визуальный элемент интерфейса МП) отвечает за верхнюю часть экрана, в котором расположены наименование вкладки и кнопки, отвечающие за масштаб и ориентацию карты, а также выпадающий список, из которого можно выбрать карты для просмотра. Во второй *UI*-компонент, имеющий табличную структуру, вложена группа *UI*-компонентов. В первой строке таблицы расположены два текстовых поля с выпадающим списком, с помощью которых пользователь может выбрать начальную и конечную точки для прокладывания маршрута, и кнопка, отвечающая за расчёт маршрута и его отображение. Во второй строке таблицы расположен *UI*-компонент, на котором отображается карта и путь, который будет рассчитан МП.

Карты используются в автономном режиме, т.к. все необходимые данные хранятся в приложении, и расчёт оптимального пути не требует подключения к сети Интернет, а выполняется на устройстве пользователя.

Заключительная строка таблицы содержит три кнопки:

- предыдущее действие (если путь проложен не только на одном этаже, а затрагивает переходы между этажами и корпусами);
- очистить (стирает маршрут из памяти и экрана, а также блокирует соседние кнопки);
- следующее действие (если путь проложен не только на одном этаже, а затрагивает переходы между этажами и корпусами).

## 2 Описание и проверка работоспособности МП

В качестве среды для разработки МП выбрана платформа *Android Studio*<sup>1</sup>. Для разработки программного продукта использовался объектно-ориентированный язык *Flutter*<sup>2</sup>. Выбранный язык является кроссплатформенным языком, что позволяет создавать МП под *Android* и *iOS*, а также для решений под *Windows*, *macOS* и *Linux* с использованием языка

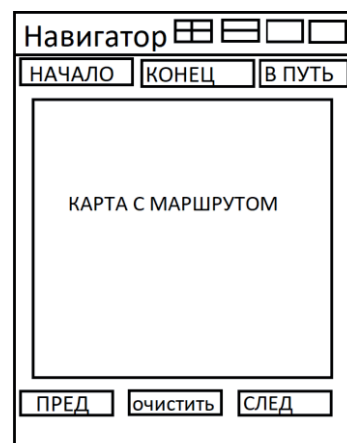


Рисунок 1 – Вкладка «Навигация» МП

<sup>1</sup> Android Studio. <https://developer.android.com/studio>.

<sup>2</sup> Flutter. <https://flutter.dev/>.

программирования *Dart*<sup>3</sup>. В языке *Dart* нет готового инструмента для отображения карт, но с помощью доступного функционала можно создать новый функционал, отвечающим заданным требованиям.

Для разработки МП был создан шаблон данных, который отображается на экране устройства. С помощью встроенного во фреймворке *Flutter* инструмента, в котором реализован функционал отображения примитивных геометрических фигур (точка, линия, окружность) и окрашивания их в любой цвет, написан класс, обрабатывающий входной многомерный массив данных и отображающий результат на карте. На рисунке 2 приведён пример шаблона данных, в котором:

- массив точек карты – стены здания;
- массив точек карты – стены внутренней планировки помещений в здании;
- массив точек графа – дороги (возможные маршруты);
- массив точек легенды карты – местонахождение пронумерованных аудиторий (корпусов и др.);
- таблица смежности графа – указание путей между аудиториями (корпусами и др.), являющихся вершинами графа.

```
[
  [массив точек карты],
  [массив точек карты],
  [массив точек графа],
  [массив точек легенды карты],
  [таблица смежности графа],
]
```

Рисунок 2 – Шаблон данных

На рисунке 3 показан пример шаблона данных МП, принимающего входные данные и отображающего возможные маршруты внутри помещения, с нумерацией аудиторий. Для разработки программного обеспечения выполнена предварительная работа на базе предоставленных ОГУ документов о планировке корпусов и актуальной информации о расположении аудиторий и иных помещений. Подготовленный материал форматирован к виду, соответствующему шаблону данных. Полученный результат помещён в БД, которая размещена на удалённом сервере. В интерфейсе поля, отвечающего за выбор начальной и конечной точек маршрута, происходит наполнение информацией из БД, т.е. из полей легенды карты при старте программы. Когда пользователь совершает выбор, то данные записываются во временные переменные, описанные в логической архитектуре, представляющей собой объектно-ориентированную программу, которая содержит слои, классы объектов, пакеты, основные платформы, интерфейсы, подсистемы и их взаимодействия.

На рисунке 4 приведён пример выбора начальной точки маршрута. В интерфейсе предусмотрен выпадающий список, из которого пользователь может выбрать нужную карту для отображения. Результат записывается во временную переменную, описанную в логической архитектуре, и передаётся в класс для отображения. Пример выпадающего списка представлен на рисунке 5.

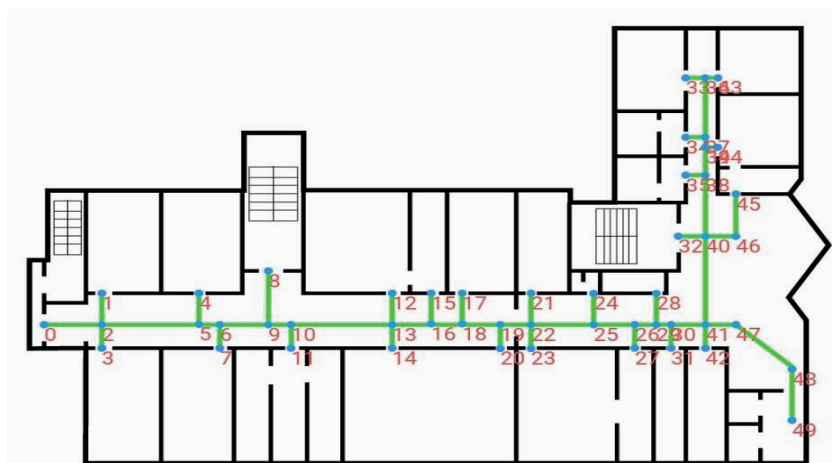


Рисунок 3 – Пример шаблона данных мобильного приложения

<sup>3</sup> <https://dart.dev/>.

Для проверки работоспособности МП проведён эксперимент, согласно которому требуется построить кратчайший путь от 16-го корпуса до аудитории 20119, расположенной в 20-ом корпусе на 1-ом этаже. На рисунке 6 представлена карта результата эксперимента.



Рисунок 4 – Пример выбора начальной точки маршрута

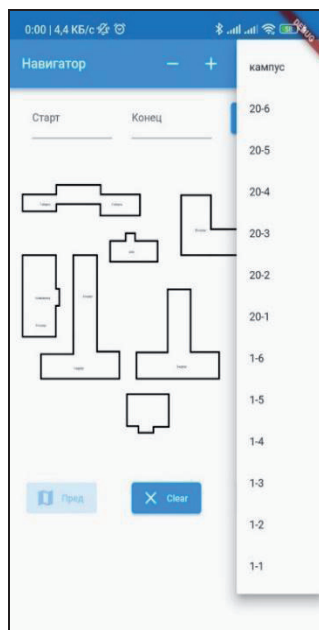
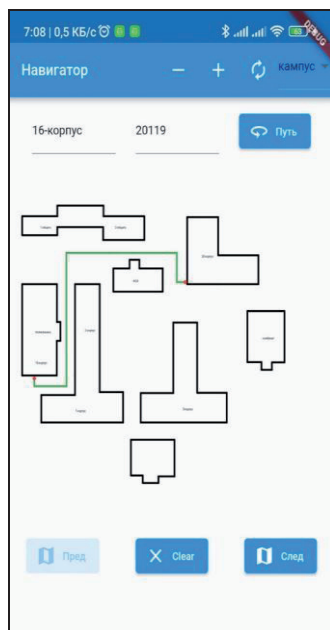


Рисунок 5 – Пример выпадающего списка



а – маршрут между корпусами; б – маршрут в корпусе

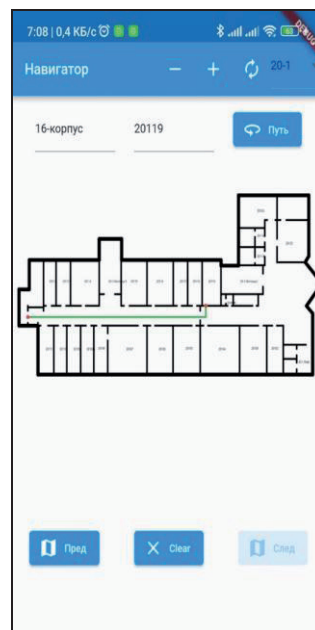


Рисунок 6 – Карта результата эксперимента

На рисунке 7 приведён построенный МП маршрут, на котором числа означают последовательность точек графа, через которые проходит путь.

```
I/flutter (18921): Начало маршрута: 16-корпус зайдите в: 20-корпус
I/flutter (18921): Маршрут: 26 25 27 28 30 32 2 3 6 7 8
I/flutter (18921): идите от входа до конца маршрута: 20119
I/flutter (18921): Маршрут: 8 2 3 6 7 9 11 13 16 17 19 21 24 26 29 31 33
```

Рисунок 7 – Построенный мобильным приложением маршрут эксперимента

Для упрощения добавления новых карт в БД МП, разработан вспомогательный сервис, написанный на языке программирования PHP<sup>4</sup>, который преобразует SVG картинку<sup>5</sup> в набор данных по созданному шаблону.

SVG-изображение имеет следующую структуру:

- 1 слой – стены здания;
- 2 слой – стены внутренней планировки помещений;
- 3 слой – дороги (возможные маршруты) ;
- 4 слой – местонахождение точек графа (окружности малого диаметра);
- 5 слой – местонахождение пронумерованных аудиторий, корпусов и иных помещений (окружности большого диаметра).

<sup>4</sup> Lerdorf, Rasmus (June 8, 1995). Announce: Personal Home Page Tools (PHP Tools). Retrieved 7 June 2011.

<sup>5</sup> SVG (от *Scalable Vector Graphics* — масштабируемая векторная графика) — это вид графики, которую создают с помощью математического описания геометрических примитивов, которые и образуют все детали изображения.

На рисунке 8 представлен пример изображения карты возможных маршрутов, которая обрабатывается вспомогательным сервисом.

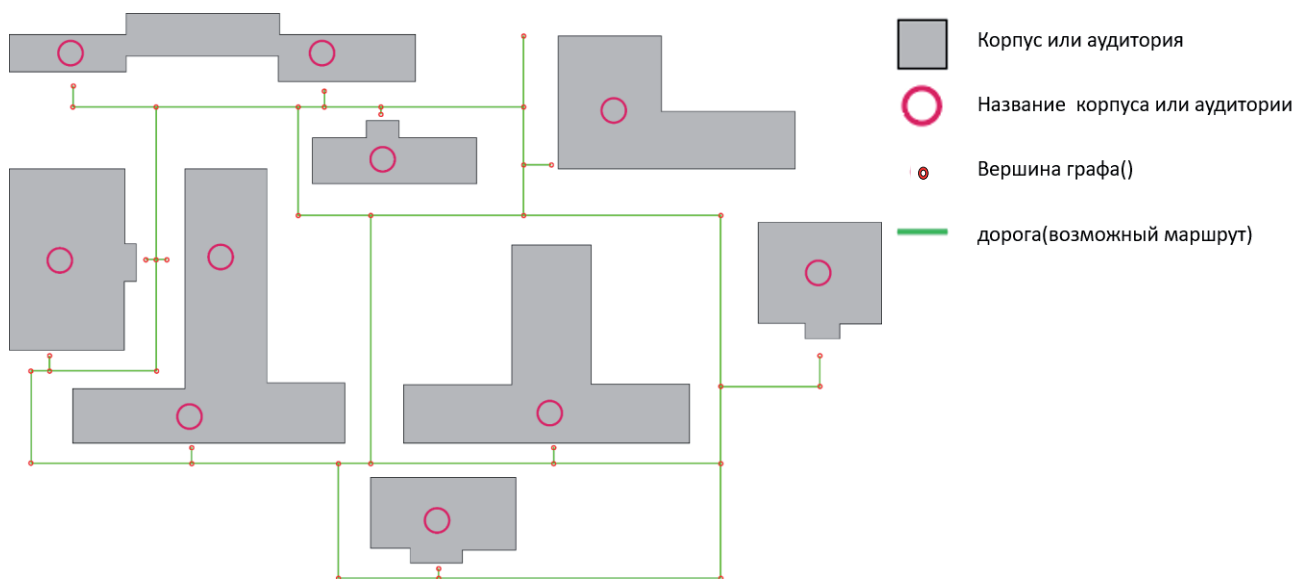


Рисунок 8 – Карта возможных маршрутов

Сервис поочередно обрабатывает слой и выдает данные согласно шаблону, после чего они интегрируются в МП.

На рисунке 9 приведена структура SVG-изображения, полученная за счёт обработки данных о планировке помещений и актуальной информации. На основании слоёв 3 и 4 за счёт работы скрипта PHP генерируется таблица смежности графа, приведённая на рисунке 10. На вход подаются данные, представленные на рисунке 9, вычисляются расстояния между точками, на основании бинарных отношений принадлежности точек прямым и пересечений прямых определяются ближайшие соседи для всех вершин (рассматривает маршруты, проходящие через окружности из слоя 4 и отыскиваются ближайшие окружности, через которые проходят эти дороги).

```
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="196.5" cy="168.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="195.5" cy="205.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="342.5" cy="205.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="342.5" cy="474.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="324.5" cy="474.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="361.5" cy="474.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="343.5" cy="670.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="194.5" cy="670.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="154.5" cy="643.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="121.5" cy="670.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="121.5" cy="832.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="405.5" cy="833.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="405.5" cy="805.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="664.5" cy="833.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="721.5" cy="833.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="664.5" cy="1035.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="841.5" cy="1035.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="841.5" cy="1018.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1339.5" cy="1035.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1339.5" cy="833.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1044.5" cy="833.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1044.5" cy="805.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="721.5" cy="396.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="593.5" cy="396.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="991.5" cy="396.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="991.5" cy="307.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1040.5" cy="307.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="991.5" cy="205.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="739.5" cy="205.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="739.5" cy="218.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="639.5" cy="205.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="639.5" cy="177.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="593.5" cy="205.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="991.5" cy="80.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1339.5" cy="396.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1339.5" cy="697.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1514.5" cy="697.5" r="3.5"/>
<circle style="fill:none;stroke:#FF0000;stroke-width:2;stroke-miterlimit:10;" cx="1514.5" cy="643.5" r="3.5"/>
```

Рисунок 9 – Структура SVG-изображения

Table with a single column containing a large grid of binary values (0s and 1s), representing an adjacency matrix for a graph. The values are arranged in a regular grid pattern, with some 1s indicating connections between nodes.

Рисунок 10 – Таблица смежности

Заключительным этапом работы вспомогательного сервиса является выгрузка конечного результата в БД, которая расположена на удалённом сервере.

## Заключение

Разработано МП для поиска оптимального маршрута от начальной точки до точки назначения с функционалом, призванным создать благоприятные условия для удобной навигации в университете. Новизна предложенной разработки заключается в универсальности средств отображения, добавления и редактирования карт. Дальнейшее совершенствование МП возможно в направлении улучшения его точности за счёт объединения алгоритмов поиска путей в графе, а также перевод карт из 2D- в 3D-пространство.

Представленные результаты исследования могут быть использованы при разработке МП для поиска пешеходных маршрутов в других университетах и учреждениях.

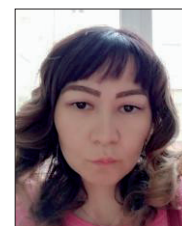
## Список источников

- [1] Официальный сайт Оренбургского государственного университета. <http://www.osu.ru/>.
- [2] **Жаркова С.А.** Актуальность разработки мобильных приложений для Android // Научные исследования в современном мире: опыт, проблемы и перспективы развития: сборник научных статей по материалам IX Международной научно-практической конференции, 18 ноября 2022. С.215-220.
- [3] Яндекс Карты [Электронный ресурс]. URL: <https://mobile.yandex.ru/apps/android/maps>.
- [4] 2GIS [Электронный ресурс]. URL: <https://info.2gis.com/>.
- [5] Google Maps [Электронный ресурс]. URL: <https://bestmaps.ru/google-maps>.
- [6] HERE WeGo [Электронный ресурс]. URL: <https://www.here.com/>.
- [7] Maps.me [Электронный ресурс]. URL: <https://maps.me/>.
- [8] Sygic GPS Navigation [Электронный ресурс]. URL: <https://www.sygic.com/>.
- [9] Magic Earth Navigation & Maps [Электронный ресурс]. URL: <https://www.amazon.com/Magic-Earth-Navigation-and-Maps/dp/B077XTY4F9>.
- [10] iGO Navigation [Электронный ресурс]. URL: <https://www.malavida.com/en/soft/igo-navigation/android/>.
- [11] OsmAnd [Электронный ресурс]. URL: <https://osmand.net/>.
- [12] MapFactor Navigation [Электронный ресурс]. URL: <https://navigatorfree.mapfactor.com/en/>.
- [13] **Алексеев В.Б.** Дискретная математика. Москва: ИНФРА-М, 2023. 133 с.
- [14] **Редькин Н.П.** Дискретная математика. Москва: ФИЗМАТЛИТ, 2009. 264 с.
- [15] **Плотников О.А., Подвальный Е.С.** Решение задачи поиска оптимального пути между двумя точками на графе с нерегулярным весом ребер // Вестник ВГТУ. 2012. № 6. С. 22-26.
- [16] **Бойков В.А.** О применении жадных алгоритмов в некоторых задачах дискретной математики // Программные продукты и системы. 2019. №1. С. 55-62.
- [17] **Ларкман К.** Применение UML и шаблонов проектирования М. Издательский дом «Вильямс», 2004. 624 с.
- [18] **Самуйлов С.В.** Объектно-ориентированное моделирование на основе UML. Саратов: Вузовское образование, 2016. 137 с.
- [19] ГОСТ 701-90 (ИСО 5807-85) Единая система программной документации (ЕСПД). Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения. М.: Стандартинформ, 2010. 23 с.

---

## Сведения об авторах

**Дусакаева Слушаш Тугайбаевна**, 1977 г. рождения. Окончила с отличием физико-математический факультет Оренбургского государственного педагогического университета им. В.П. Чкалова в 1999 г., к.т.н. (2019). Доцент кафедры прикладной математики ОГУ. В списке научных трудов более 40 работ в области системного анализа, систем поддержки принятия решений, теории нечётких множеств и систем. Author ID (РИНЦ): 673892. ORCID: 0000-0002-5292-1114. [slushashdusakaeva@rambler.ru](mailto:slushashdusakaeva@rambler.ru).✉







**Савинов Виталий Вячеславович**, 2000 г. рождения. Окончил факультет математики и информационных технологий ОГУ по направлению подготовки Прикладная математика и информатика в 2022 г. Магистрант института математики и информационных технологий ОГУ. [savinovvet@gmail.com](mailto:savinovvet@gmail.com).

Поступила в редакцию 10.07.2023, после рецензирования 14.08.2023. Принята к публикации 23.08.2023.



Scientific article

DOI: 10.18287/2223-9537-2023-13-3-455-464

## Mobile application for finding the best route on campus

© 2023, S.T. Dusakaeva ✉, V.V. Savinov

Orenburg State University, Orenburg, Russia

### Abstract

The actual task of developing a mobile application for finding the optimal route on a university campus is considered. The mathematical basis for designing a mobile application is the task of finding the shortest route in a graph whose vertices are classrooms, laboratories, libraries, departments, dean's offices, etc. in the buildings of the university. The analysis of the main characteristics of existing mobile applications for walking navigation has been carried out, their advantages and disadvantages have been identified. The well-known algorithms for finding the shortest path in a graph are considered: breadth-first search, greedy algorithm, and Dijkstra shortest path algorithm. For the task under consideration, a software implementation and an auxiliary service have been developed to create maps and bring them to a format in which a graph is automatically created for the correct operation of the application. The novelty of the proposed development lies in the possibility of displaying, adding and editing various maps. Further improvement of the developed mobile application can be carried out in the direction of increasing the accuracy of navigation by combining algorithms for finding paths in a graph, as well as in translating maps from 2D to 3D. The results of the study can be used to find the best routes in other universities.

**Key words:** map navigation, mobile application, optimal route, algorithm, finding the shortest path in a graph, campus.

**For citation:** Dusakaeva ST, Savinov VV. Mobile application for finding the best route on campus [In Russian]. *Ontology of designing*. 2023; 13(3): 455-464. DOI: 10.18287/2223-9537-2023-13-3-455-464.

**Conflict of interest:** The authors declare no conflict of interest.

### List of figures and tables

Figure 1 – Navigation tab

Figure 2 – Data Template

Figure 3 – Example of data template

Figure 4 – Example of choosing the starting point of a route

Figure 5 – Example of a drop-down list

Figure 6 – Experiment result map

Figure 7 – Recommended route of the experiment

Figure 8 – Map of possible routes

Figure 9 – SVG image Structure

Figure 10 – Adjacency table

Table 1 – Main characteristics and advantages

Table 2 – Comparative analysis of algorithms for finding the shortest path in a graph)

## References

- [1] Official website of Orenburg State University. <http://www.osu.ru/>.
- [2] **Zharkova SA.** Relevance of the development of mobile applications for Android [In Russian]. Scientific research in the modern world: experience, problems and prospects of development: collection of scientific articles based on the materials of the IX International Scientific and Practical Conference, November 18, 2022. P.215-220.
- [3] Yandex Maps [Electronic resource]. URL: [mobile.yandex.ru/apps/maps](http://mobile.yandex.ru/apps/maps).
- [4] 2GIS [Electronic resource]. URL: [info.2gis.com](http://info.2gis.com).
- [5] Google Maps [Electronic resource]. URL: <https://bestmaps.ru/google-maps>.
- [6] HERE WeGo [Electronic resource]. URL: [here.com](http://here.com).
- [7] Maps.me [Electronic resource]. URL: [maps.me](http://maps.me).
- [8] Sygic GPS Navigation [Electronic resource]. URL: <https://www.sygic.com/>.
- [9] Magic Earth Navigation & Maps [Electronic resource]. URL: <https://www.amazon.com/Magic-Earth-Navigation-and-Maps/dp/B077XTY4F9>.
- [10] iGO Navigation [Electronic resource]. URL: <https://www.malavida.com/en/soft/igo-navigation/android/>.
- [11] OsmAnd [Electronic resource]. URL: [osmand.net](http://osmand.net).
- [12] MapFactor Navigator [Electronic resource]. URL: [navigatorfree.mapfactor.com](http://navigatorfree.mapfactor.com).
- [13] **Alekseev VN.** B. Discrete mathematics. [In Russian]. Moscow: INFRA-M, 2023. 133 P.
- [14] **Redkin NP.** Discrete mathematics. [In Russian]. Moscow: FIZMATLIT, 2009. 264 p.
- [15] **Plotnikov OA, Podvalny ES.** Solving the problem of finding the optimal path between two points on a graph with an irregular edge weight [In Russian]. Vestnik VSTU. 2012; 6: 22-26.
- [16] **Boikov VA.** On the application of greedy algorithms in some problems of discrete mathematics [In Russian]. Software products and systems. 2019; 1: 55-62.
- [17] **Larkman K.** Application of UML and design patterns [In Russian]. Moscow: Publishing house "Williams", 2004. 624 p.
- [18] **Samuilov SV.** Object-oriented modeling based on UML [In Russian]. Saratov: University Education, 2016. 137 p.
- [19] GOST 701-90 (ISO 5807-85) Unified system of Software Documentation (USPD) [In Russian]. Schemes of algorithms, programs, data and systems. Conditional designations and rules of execution. Moscow: Standartinform, 2010. 23 p.

---

## About the authors

**Slushash Tugaybayevna Dusakaeva**, (b. 1977) graduated with honors from the Faculty of Physics and Mathematics of the Orenburg State Pedagogical University named after V.P. Chkalov in 1999, Candidate of Technical Sciences (2019). Associate Professor of the Department of Applied Mathematics of Orenburg State University. The list of scientific papers includes more than 30 works in the field of system analysis, decision support systems, theory of fuzzy sets and systems. Author ID (RSCI): 673892. ORCID: 0000-0002-5292-1114 [slushashdusakaeva@rambler.ru](mailto:slushashdusakaeva@rambler.ru). ✉.

**Vitaly Vyacheslavovich Savinov** (b. 2000) graduated from the Faculty of Mathematics and Information Technologies of Orenburg State University in the direction of Applied Mathematics and Computer Science in 2022. Master's student of the Institute of Mathematics and Information Technologies of Orenburg State University. [savinovvet@gmail.com](mailto:savinovvet@gmail.com).

---

Received July 10, 2023. Revised August 14, 2023. Accepted August 23, 2023.

---