1

Санкт-Петербургский политехнический университет Институт компьютерных наук и технологий

**Кафедра «Информатика и вычислительная техника»**

**КУ Р С О В ОЙ П Р О Е К Т**

**Разработка приложения «Навигатор»**

по дисциплине «Технологии программирования»

Выполнил студент

гр. 3530901/20002 Ермолин А.А

Преподаватель Степанов Д.С.

26 мая 2023 г.

Санкт-Петербург

2023

Санкт-Петербургский политехнический университет

**ЗАДАНИЕ**

**НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА**

студенту группы 3530901/20002 Ермолину Андрею Алексеевичу

1. Тема проекта: создание приложения «навигатор» для навигации и поиска пути в третьем корпусе Политехнического университета.
2. Срок сдачи законченного проекта:26 мая
3. Исходные данные к проекту: требования к реализовываемому проекту
4. Содержание пояснительной записки: введение с описанием правил игры, основная часть (технологии jetpack compose, их применение в приложении), описание автоматических тестов для бизнес-логики, заключение, список используемых источников.

Дата получения задания: «3» марта 2023 г.

Руководитель Степанов Д.С.

Задание принял к исполнению Ермолин А. А.

3 марта 2023

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_TOC_250004)

[ОПИСАНИЕ ПРЕДЛОЖЕННОГО РЕШЕНИЯ 5](#_TOC_250003)

[ТЕСТИРОВАНИЕ ПРОГРАММЫ 6](#_TOC_250002)

[ЗАКЛЮЧЕНИЕ 7](#_TOC_250001)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 8](#_TOC_250000)

# ВВЕДЕНИЕ

Цель работы: создать и протестировать приложение «Навигатор» с графическим интерфейсом.

Поставленная задача:

1. Для начала поиска пути стоит определить текущее местоположение объекта в корпусе. Это я могу сделать при помощи GPS или используя силу сигнала Wifi-роутеров. Я выбрал второй вариант, так как использование GPS внутри здания - не лучшая идея.
2. Строим граф описания корпуса, создаем функцию поиска кратчайшего пути между нодами.
3. Получив начальное местоположение и присоединив его к ближайшей ноде находим путь до точки, выбранной пользователем, отображаем на экране.

# ОПИСАНИЕ ПРЕДЛОЖЕННОГО РЕШЕНИЯ

Для создания графического пользовательского интерфейса (GUI) использовалась библиотека jetpack compose, которая обладает большим числом заготовок элементов интерфейса. В частности были использованы следующие элементы: кнопка (Button), текстовое поле (TextField) а также вертикальные и горизонтальные ряды (Column и Row). Взаимодействие пользователя с графическим элементом описывается в событиях для этого элемента.

В классе Valuable содержится граф, а также результаты сканирования всех нод графа, использующиеся для последующего определения местоположения

Класс MainActivity содержит основные функции программы, такие как нахождение ближайшей ноды по результатам сканирования и нахождение пути. Также здесь описывается графическое представление графа на экране устройства и выбор пути пользователем

Класс SplashScreen содержит описание Activity для загрузки приложения, которая вызывается при запуске и затем переносит нас уже в класс MainActivity.

# ТЕСТИРОВАНИЕ ПРОГРАММЫ

Для тестирование бизнес-логики было написано 4 автоматических теста, использующих возможности библиотеки JUnit.

Первый тест является стандартным в Андроид разработке и проверяет правильность указанного Context’a. Второй тест проверяет работоспособность функции определения текущего этажа по ближайшей ноде . Третий тест проверяет, что функция определения ближайшей ноды выдаёт стандартное местоположение(«вход») при пустом сканировании. Четвертый тест находит путь с помощью функции “findTheWay” и проверяет, чтобы он был кратчайшим.

# ЗАКЛЮЧЕНИЕ

Было создано приложение с графическим интерфейсом, предназначенное для навигации по корпусу. Также были разработаны автоматические тесты для проверки работоспособности кода. В ходе выполнения этого задания мною была использована библиотека jetpack compose.

Исходные файлы приложения лежат в репозитории на GitHub: https://github.com/Irregulariti/Navirrgator

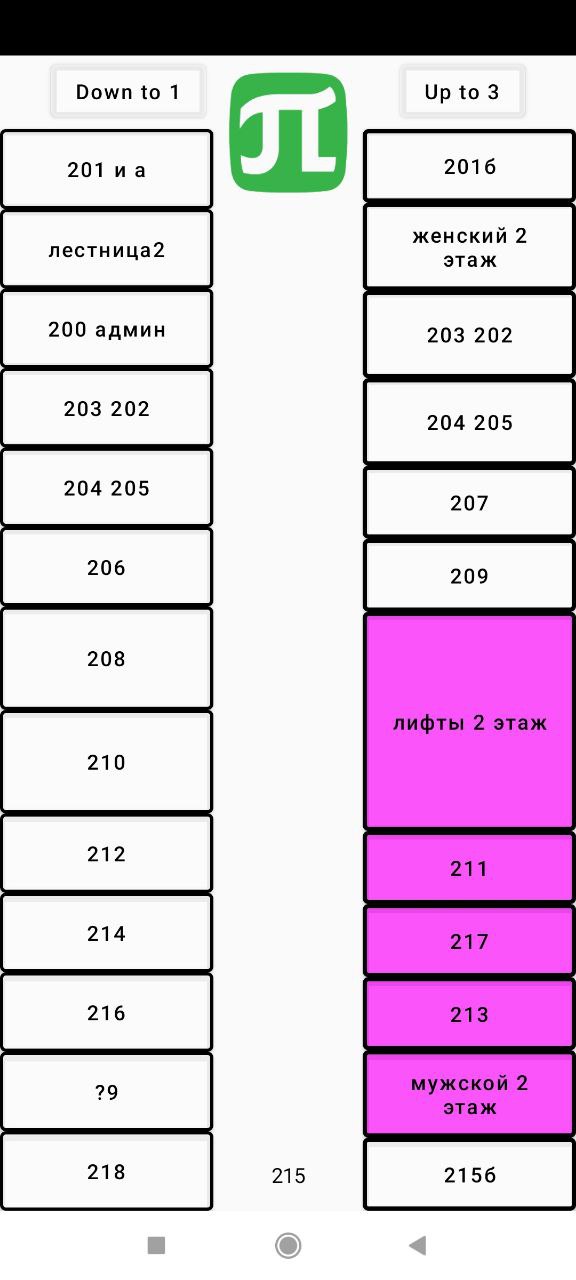


Рис 1. Скриншоты приложения

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. <https://developer.android.com/jetpack/compose> - android guide
2. <https://habr.com/ru/companies/rncb/articles/669374/> - start with compose
3. <https://metanit.com/kotlin/jetpack/1.1.php> - русский гайд на compose