|  |  |
| --- | --- |
| Название | Лабиринт |
| Description | Пользовательская веб-версия приложения «Лабиринт» |
| Версия документа | 1.0 |
|  |  |
| Автор | Сафронов Денис, Хуртина Татьяна |
| Группа | М20-Ш04 |

Оглавление

[**Общая информация** 3](#_Toc88394333)

[**Функциональное описание** 4](#_Toc88394334)

[**Пользовательские сценарии** 4](#_Toc88394335)

[**Техническое описание** 7](#_Toc88394336)

[**Диаграмма прецедентов** 8](#_Toc88394337)

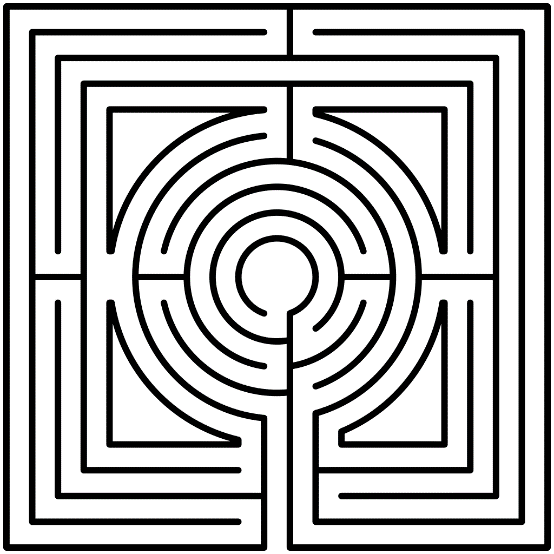
[**Диаграмма активностей** 9](#_Toc88394338)

[**Диаграмма последовательностей** 10](#_Toc88394339)

[**Диаграмма классов** 11](#_Toc88394340)

[**Приложение** 12](#_Toc88394341)

# **Общая информация**



Лабиринт

Разработано Хуртиной Татьяной, Сафроновым Денисом

Разработана для веб-версий

Дата выпуска TBD

# **Функциональное описание**

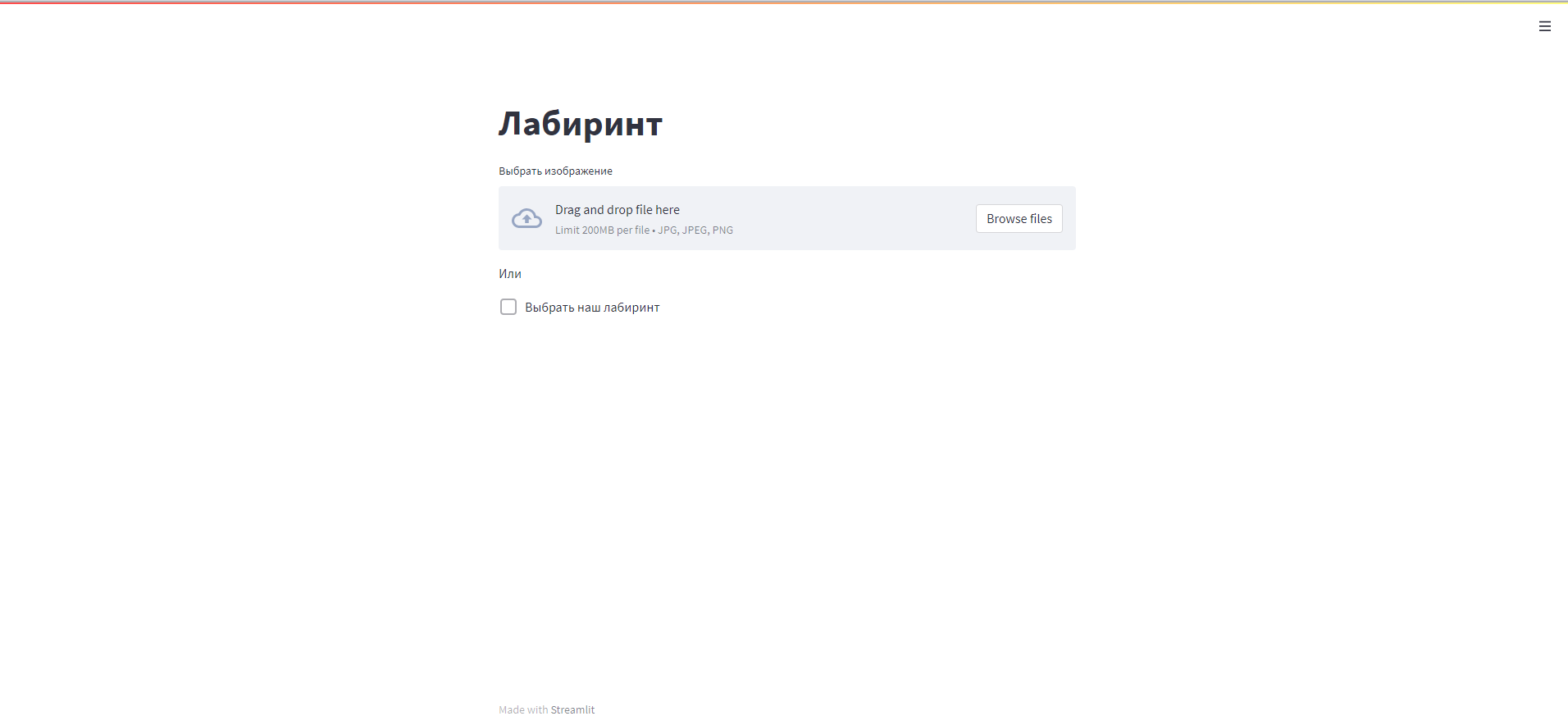
Данное приложение «Лабиринт» является игрой головоломкой, в которой пользователь может самостоятельно находить путь (решение) от точки старта до точки финиша.

В данном приложении предусмотрены возможности:

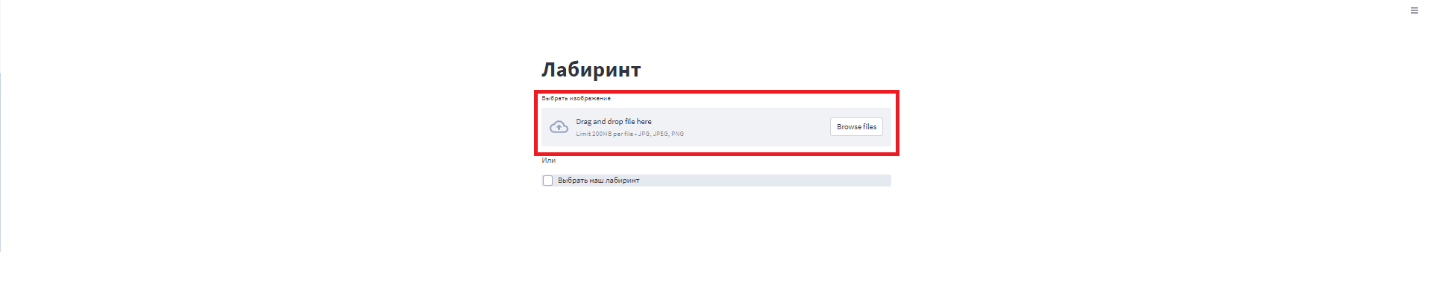
1. При попадании на главный экран предусмотрена возможность загрузки картинки с лабиринтом в различных форматах (JPG, JPEG, PNG) с размером файла не более 200Мб
2. При загрузки картинки с лабиринтом пользователю предоставляется возможность переставлять точки старта и финиша при помощи специальной панели в левой части экрана.
3. Если текущий лабиринт не подходит пользователю – есть возможность перезагрузить картинку с лабиринтом в различных форматах (JPG, JPEG, PNG) с размером файла не более 200Мб в данное приложение
4. Предоставляется пользователю возможность просмотра решения лабиринта при нажатии на кнопку «Решить лабиринт» в нижней части экрана

## **Пользовательские сценарии**

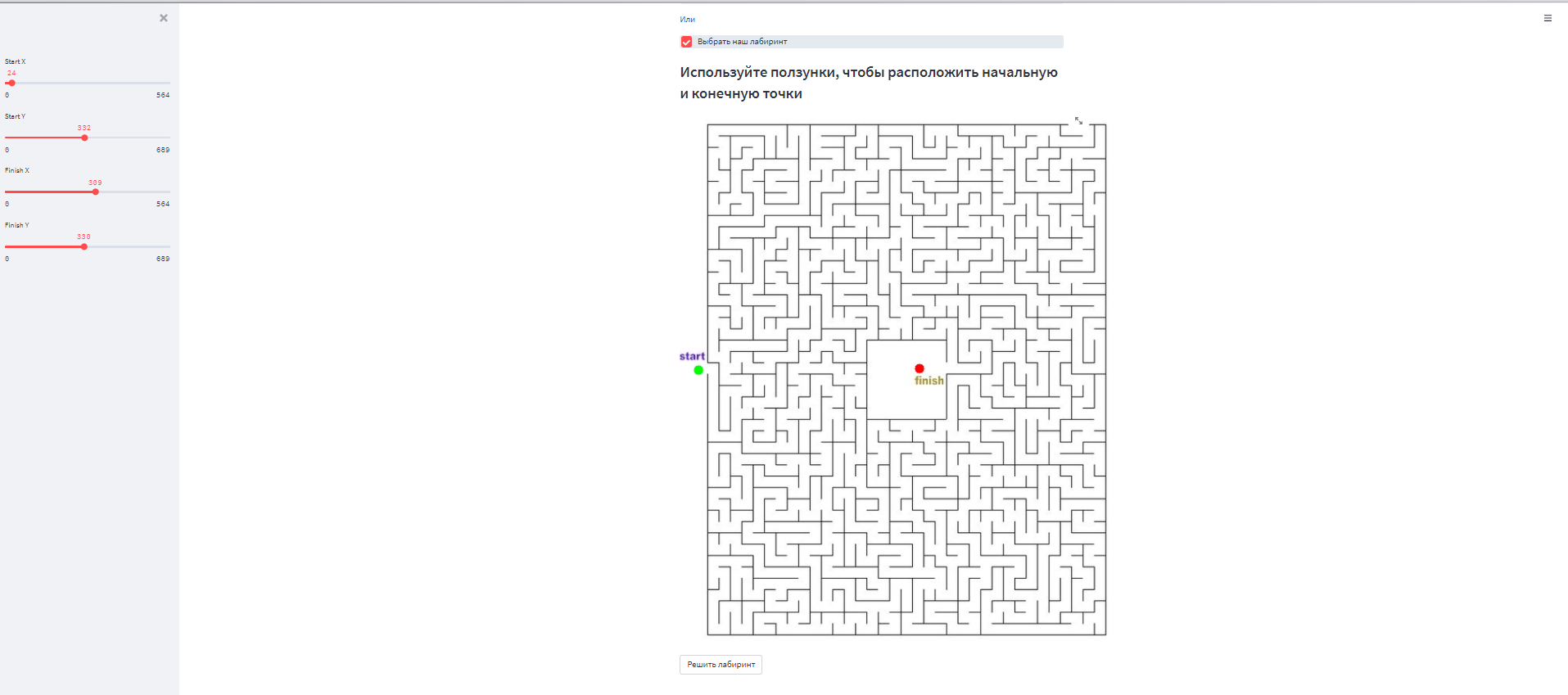
Главный экран приложения «Лабиринт»

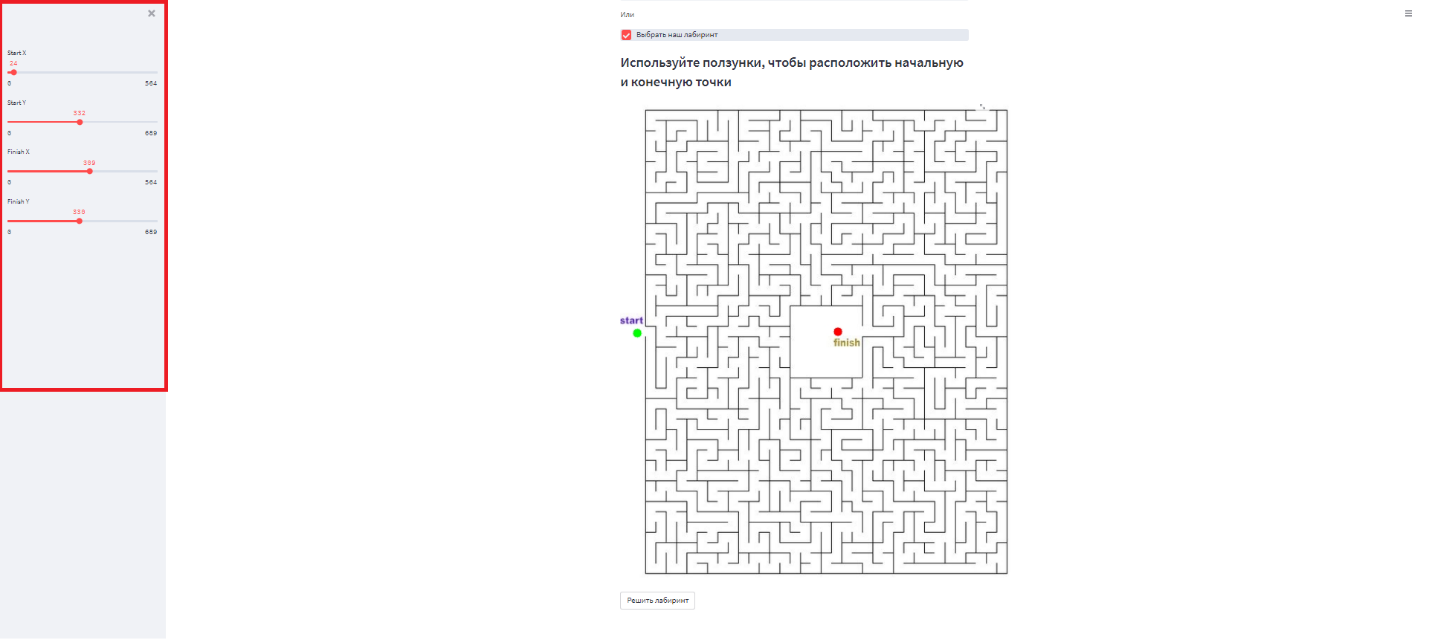
****

При нажатии на блок «Выбрать изображение» пользователю предоставляется возможность выбрать картинку с собственным лабиринтом и загрузить данную картинку в приложение «Лабиринт».

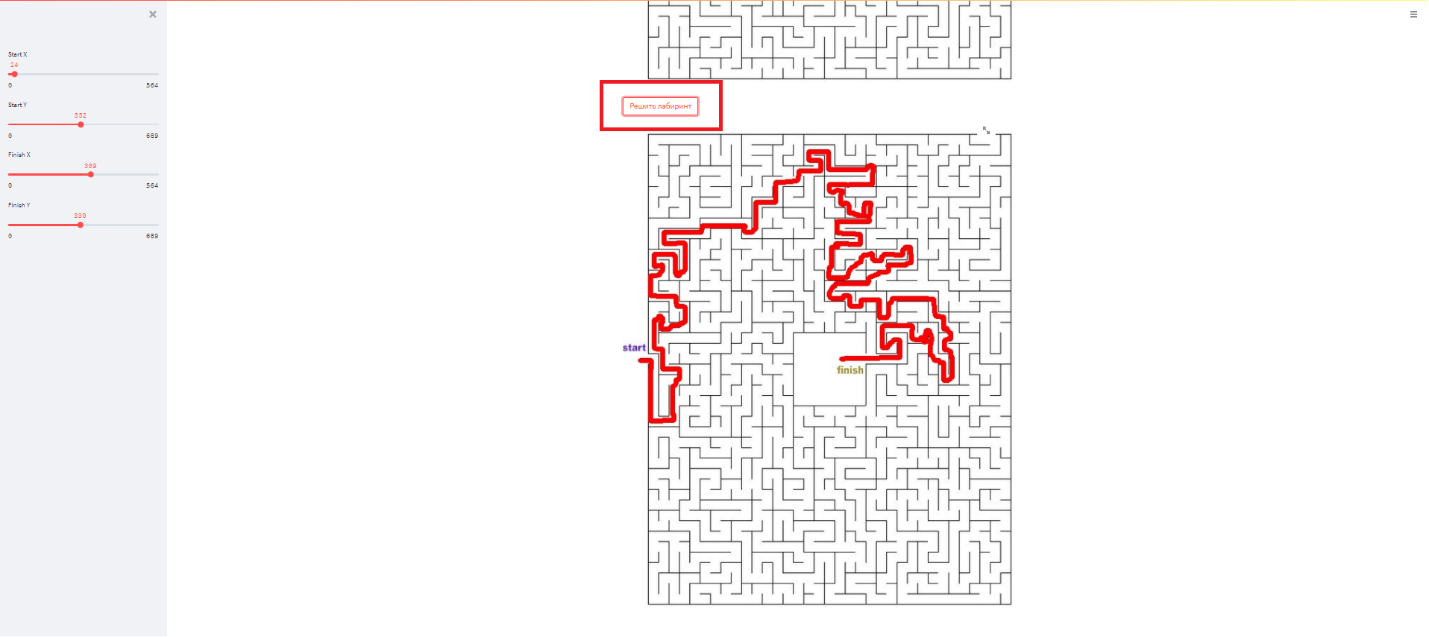


При нажатии на чек-бокс «выбрать наш вариант» пользователю предоставляется возможность начать проходить лабиринт, созданный по умолчанию



В левой части экрана расположены ползунки с возможностью выбора точек старта и финиша  


При возникновении затруднений при нахождении правильного пути от точки старта до точки финиша, в приложении предусмотрено функция «Решить лабиринт». При нажатии на кнопку «Решить лабиринт» пользователю предоставляется возможность увидеть решение данного лабиринта



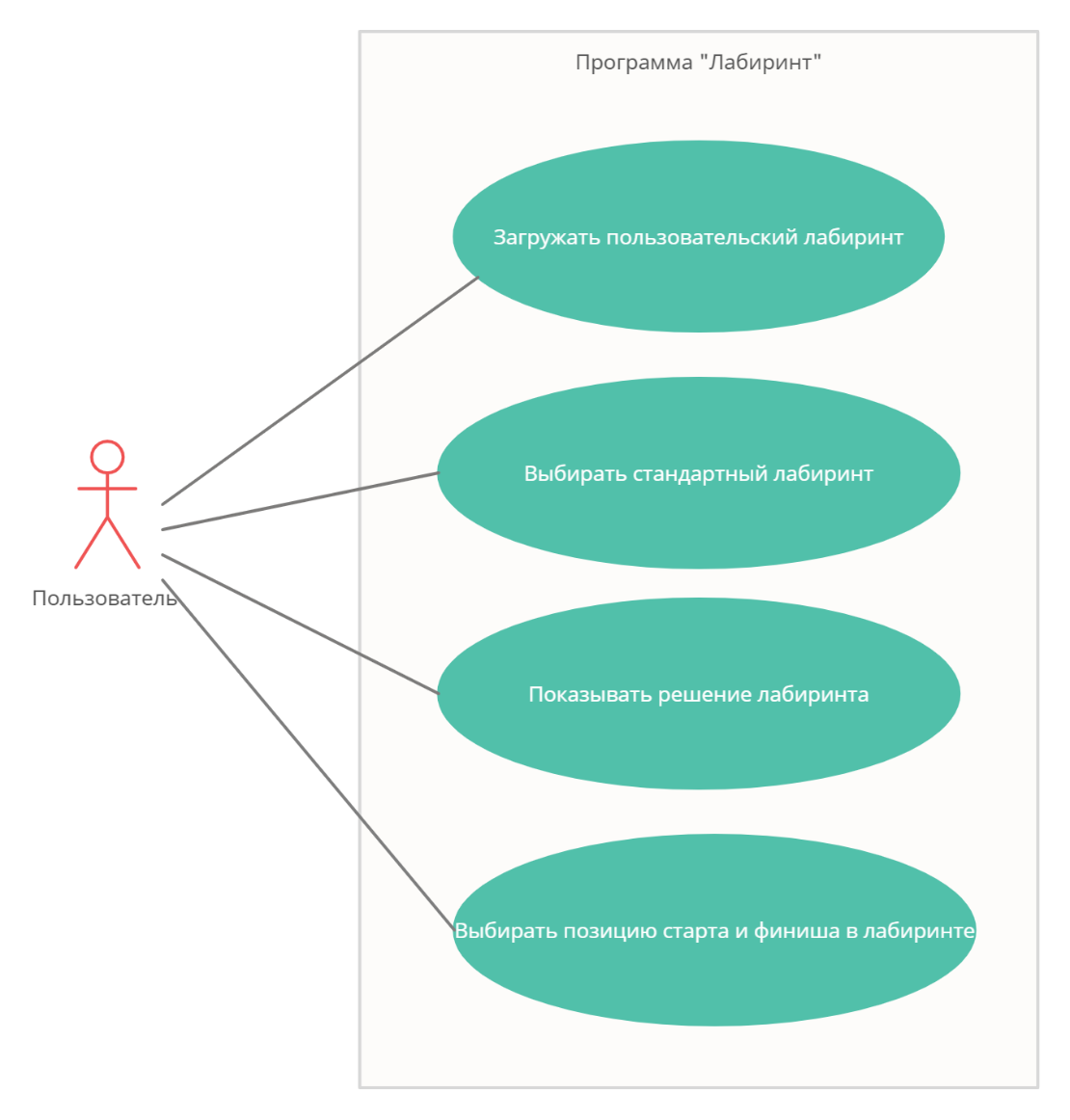
# **Техническое описание**

Приложение «Лабиринт» представляет из себя задачу программирования, в которой используются методы кратчайшего пути (алгоритмы Дейкстры описаны в главе 1 в «Приложение»).

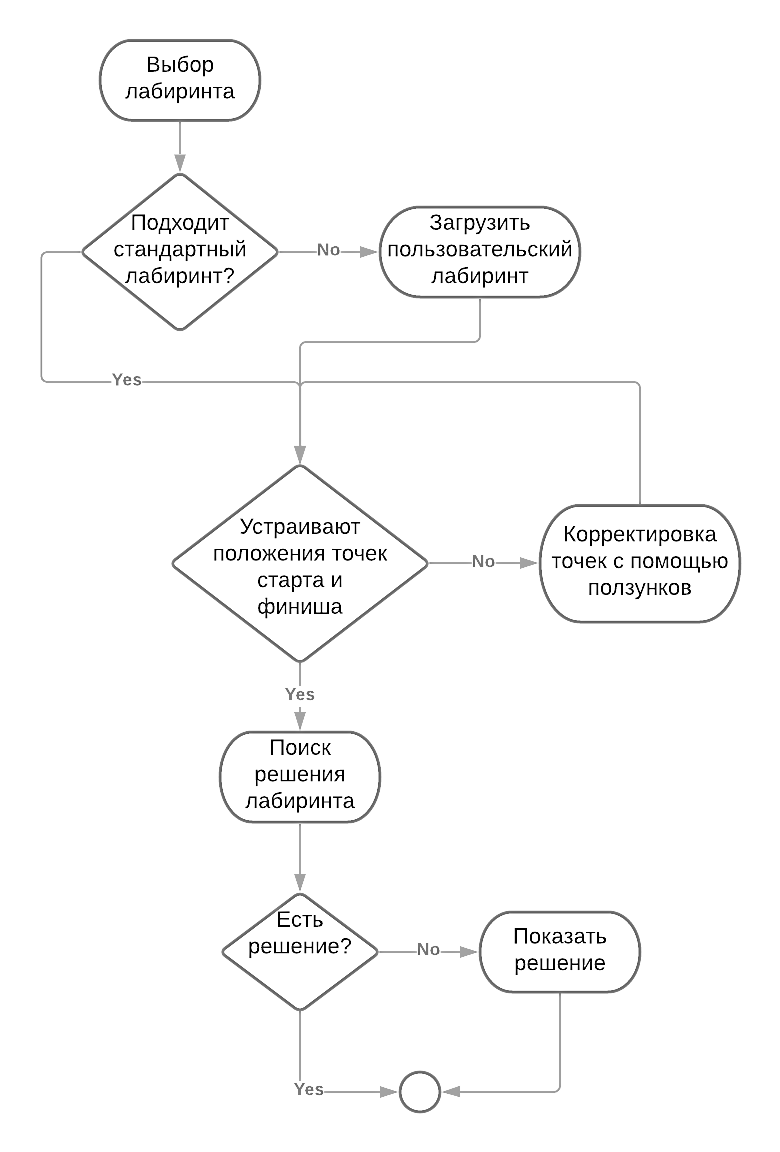
Для реализации «Лабиринт» используется язык программирования Python 3 и соответствующие библиотеки в нем:

* OpenCV - библиотека компьютерного зрения для Python, чтобы извлекать значения пикселей и показать наши изображения лабиринта
* Matplotlib - это комплексная библиотека для создания статических, анимированных и интерактивных визуализаций в Python.
* Numpy - это библиотека языка Python, добавляющая поддержку больших многомерных массивов и матриц, вместе с большой библиотекой высокоуровневых (и очень быстрых) математических функций для операций с этими массивами.
* Streamlit - это веб-фреймворк, предназначенный для исследователей данных для простого развертывания моделей и визуализаций с использованием Python

# **Диаграмма прецедентов**

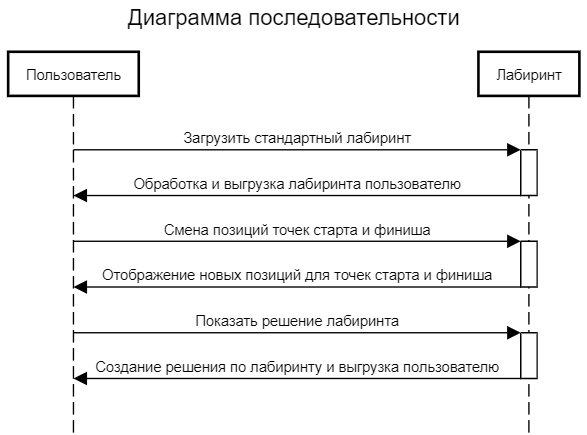


# **Диаграмма активностей**

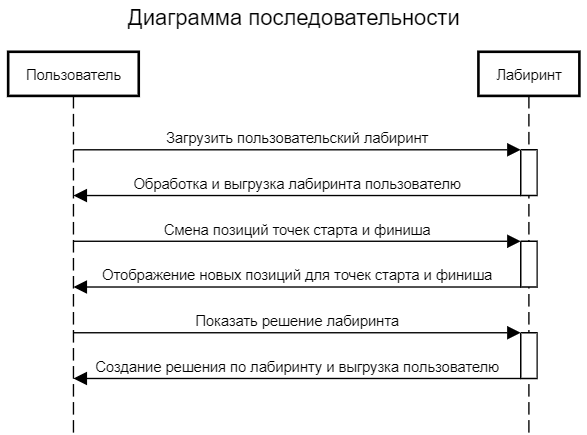


# **Диаграмма последовательностей**

Данная диаграмма рассматривает случай, когда пользователь выбирает стандартный лабиринт, заложенный в системе:

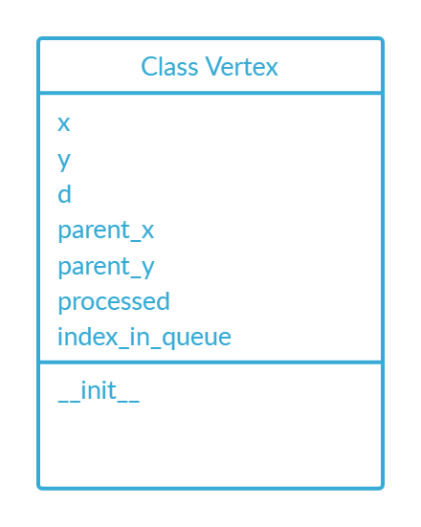


Данная диаграмма рассматривает случай, когда пользователь выбирает свой собственный лабиринт и загружает в систему:



# **Диаграмма классов**

В данном программном коде инициализирован один класс – «Vertex»



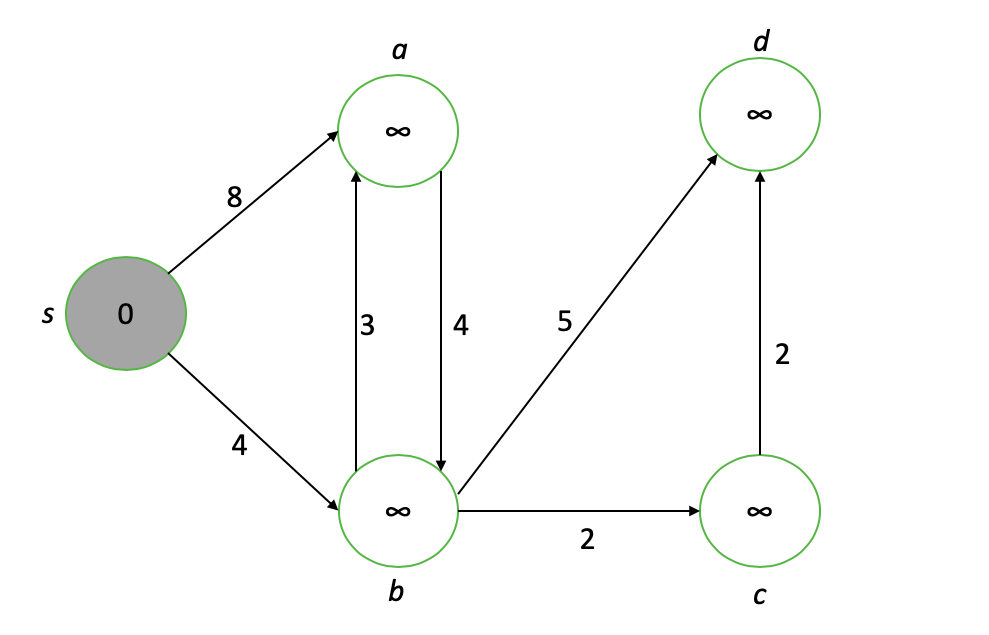
**Визуальное описание**

# **Приложение**

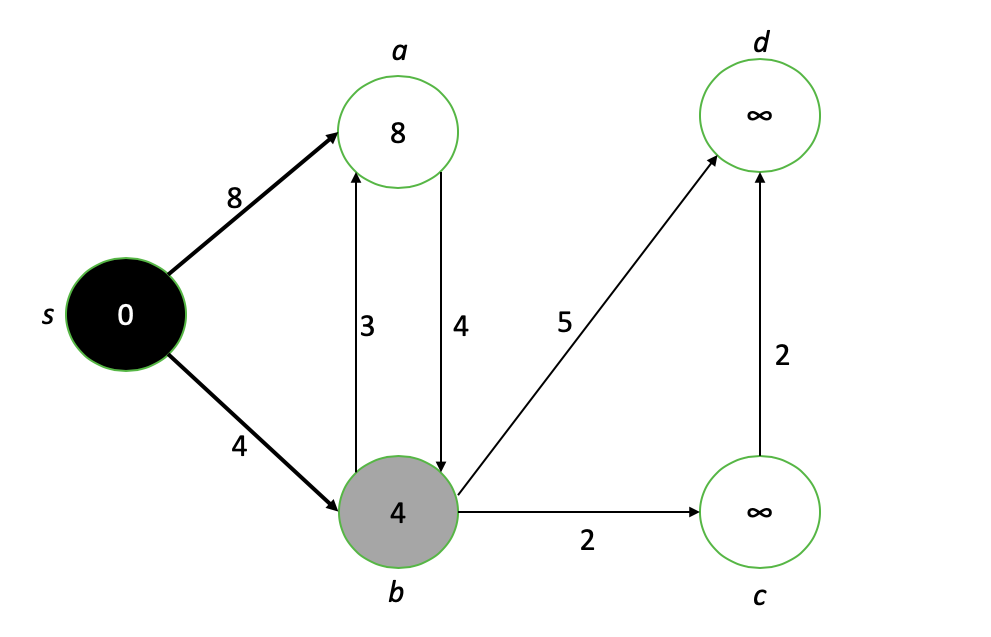
**1 . Алгоритм Дейкстры**

Алгоритм Дейкстры — один из наиболее популярных алгоритмов теории графов. Он используется для поиска кратчайшего пути между узлами на ориентированном графе. Мы начнем с исходного узла и известных длин ребер между узлами.

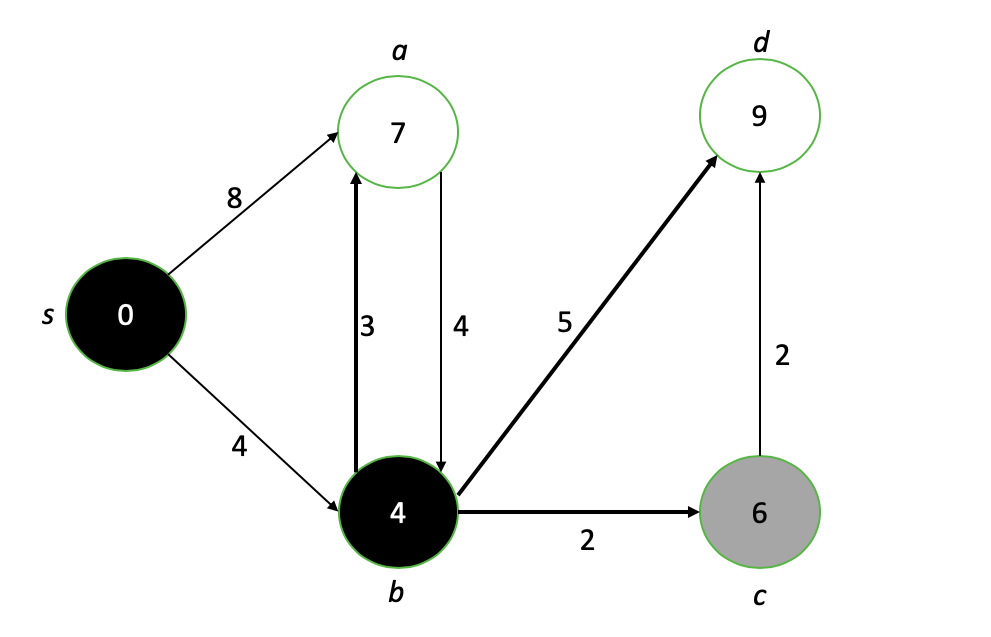
Сначала мы присваиваем значение расстояния от источника всем узлам. Узел s получает значение 0, потому что это источник; остальные получают значения ∞ для начала.



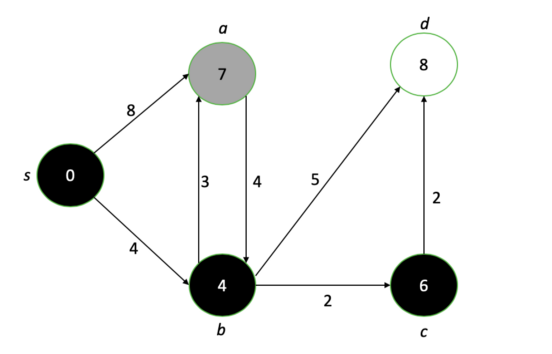
Наш интересующий узел — это необработанный узел с наименьшим значением (показан серым), то есть s. Сначала мы «ослабляем» каждую смежную вершину до нашего интересующего узла, обновляя их значения до минимума их текущего значения или значения узла интереса плюс длину соединительного ребра…

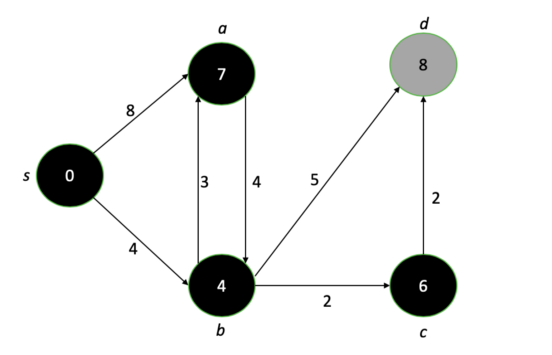


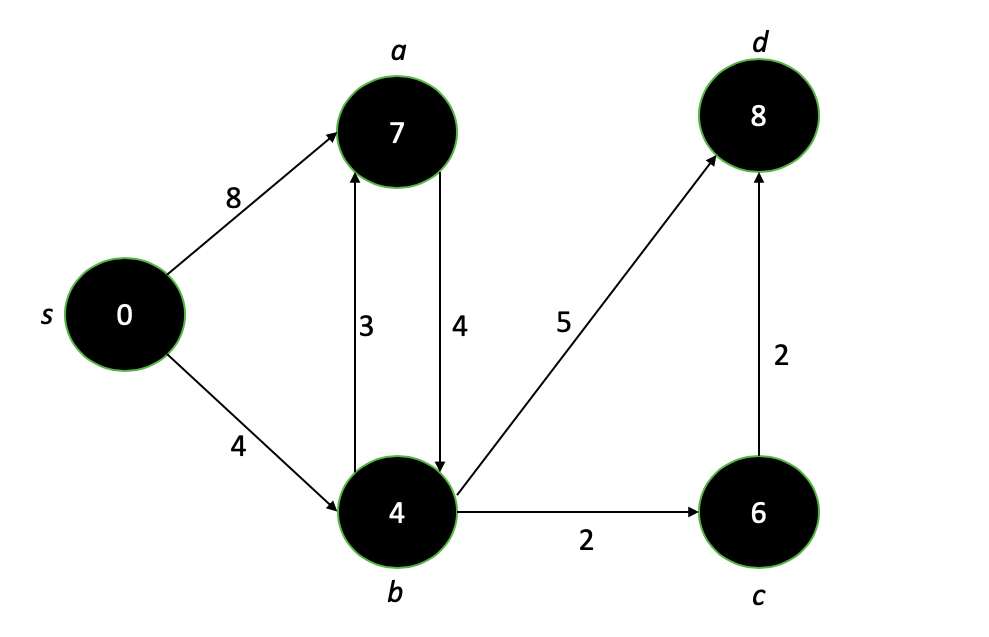
Узел s теперь завершен (черный), а его соседи a и b приняли новые значения. Новый интересующий узел — b, поэтому мы повторяем процесс «ослабления» соседних узлов b и финализации значения кратчайшего пути для b.



Пройдя через каждый узел, мы в итоге получим график, показывающий кратчайшую длину пути от источника до каждого узла.







Наша финальная диаграмма после запуска алгоритма Дейкстры. Числа в каждом узле представляют кратчайшее возможное расстояние от исходного узла.