

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий  
Кафедра «Инфокогнитивные технологии»

Направление подготовки/ специальность: 09.03.01 «Веб-технологии»

# ОТЧЕТ

по проектной практике

Студенты: Паладий Максим Юрьевич, Реброва Анастасия Алексеевна Группа:  
241-3210

Место прохождения практики: Московский Политех, кафедра  
«Инфокогнитивные технологии»

Отчет принят с оценкой \_\_\_\_\_ Дата \_\_\_\_\_

Руководитель практики: Чернова Вера Михайловна

Москва 2025

## ОГЛАВЛЕНИЕ

### ВВЕДЕНИЕ

1. Цели и задачи
2. Принцип работы технологии
3. Основные этапы разработки
4. Реализация системы
5. Индивидуальные вклады участников
6. Принцип работы системы

### ЗАКЛЮЧЕНИЕ

### СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

### ПРИЛОЖЕНИЕ

## ВВЕДЕНИЕ

Разработана интерактивная система навигации для второго корпуса МосПолитеха (ул. Прянишникова, 2А), реализующая: автоматический расчет кратчайших путей эвакуации, визуализацию маршрутов через анимацию "змейки".

Используемые технологии: HTML5, CSS3, JavaScript, алгоритм BFS

### 1. Цели и задачи

Цель: создание автоматизированной системы визуализации оптимальных путей эвакуации в режиме реального времени.

Задачи:

- 1) разработка алгоритма поиска кратчайшего пути;
- 2) создание интерактивной карты помещений;
- 3) реализация анимации движения "змейки";
- 4) интеграция с архитектурными планами здания.

### 2. Основные этапы разработки

- анализ планировки здания: изучение поэтажных планов корпуса МосПолитеха (ул. Прянишникова, 2А), выделение ключевых элементов: аудитории, коридоры, выходы, лифты, санузлы (см.

Приложение Рис.1)

- построение графа связи между помещениями (см. Приложение Рис.2)

```
const graph = {
  'exit1': ['corridor-0'],
  'exit2': ['corridor-5'],
  '2201M': ['corridor-0'],
  '2201': ['corridor-0'],
  '2203': ['corridor-1'],
  '2204': ['corridor-2'],
  '2205': ['corridor-3'],
  '2206': ['corridor-4'],
  '2214': ['corridor-0'],
  '2210': ['corridor-0'],
  '2202B': ['corridor-1'],
  '2202A': ['corridor-1'],
  'elevator': ['corridor-2'],
  '2209': ['corridor-3'],
  '2208A': ['corridor-4'],
  '2208': ['corridor-4'],
  '2207': ['corridor-5'],
  'toilet': ['corridor-5'],
  'corridor-0': ['exit1', '2201M', '2201', '2214', '2210', 'corridor-1'],
  'corridor-1': ['corridor-0', '2203', '2202B', '2202A', 'corridor-2'],
  'corridor-2': ['corridor-1', '2204', 'elevator', 'corridor-3'],
  'corridor-3': ['corridor-2', '2205', '2209', 'corridor-4'],
  'corridor-4': ['corridor-3', '2206', '2208A', '2208', 'corridor-5'],
  'corridor-5': ['corridor-4', '2207', 'toilet', 'exit2']
};
```

### 3. Реализация системы

— ядро алгоритма (BFS), который был выбран как основа для поиска кратчайшего пути эвакуации

```
function findShortestPath(start, end) {
  const queue = [[start]];
  const visited = new Set([start]);

  while (queue.length > 0) {
    const path = queue.shift();
    const node = path[path.length - 1];

    if (node === end) return path;

    (graph[node] || []).forEach(neighbor => {
      if (!visited.has(neighbor)) {
        visited.add(neighbor);
        queue.push([...path, neighbor]);
      }
    });
  }

  return null;
}
```

— визуализация маршрута (анимация движения "змейки")

```
function animateSnake() {
  if (currentStep >= currentPath.length * 10) {
    cancelAnimationFrame(animationFrame);
    return;
  }

  const progress = currentStep / (currentPath.length * 10);
  const pathIndex = Math.floor(progress * (currentPath.length - 1));
  const segmentProgress = (progress * (currentPath.length - 1)) % 1;

  const current = roomPositions[currentPath[pathIndex]];
  const next = roomPositions[currentPath[pathIndex + 1]];

  if (current && next) {
    const head = snakeSegments[snakeSegments.length - 1];
    head.style.left = `${current.x + (next.x - current.x) * segmentProgress - 7.5}px`;
    head.style.top = `${current.y + (next.y - current.y) * segmentProgress - 7.5}px`;
  }

  // Подсветка пройденного пути
  for (let i = 0; i < snakeSegments.length - 1; i++) {
    if (i < pathIndex) {
      snakeSegments[i].style.opacity = '1';
    }
  }

  currentStep++;
  animationFrame = requestAnimationFrame(animateSnake);
}
```

#### 4. Индивидуальные вклады участников

Паладий Максим Юрьевич: анализ планировки здания, разметка страницы, реализация BFS с кэшированием

Реброва Анастасия Алексеевна: добавление стилей, анимаций

#### 5. Принцип работы системы:

- 1) пользователь выбирает начальную точку;
- 2) система вычисляет путь до ближайшего выхода;
- 3) "змейка" последовательно проходит маршрут.

#### ЗАКЛЮЧЕНИЕ

Реализация проекта интерактивного плана эвакуации позволила создать комплексное решение, которое существенно превосходит традиционные системы безопасности по ключевым параметрам (быстрое построение оптимальных маршрутов, интуитивно понятная визуализация).

## СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Руководство по написанию игры «Змейка» на чистом JS – URL: <https://gamedeveloperliberant.firebaseio.com/ru/posts/snake-purejs/> (дата обращения: 13.05.2025).
2. Репозиторий с туториалами по реализации технологии – URL: <https://github.com/codecrafters-io/build-your-own-x> (дата обращения: 10.05.2025, 11.05.2025, 13.05.2025).

ПРИЛОЖЕНИЕ

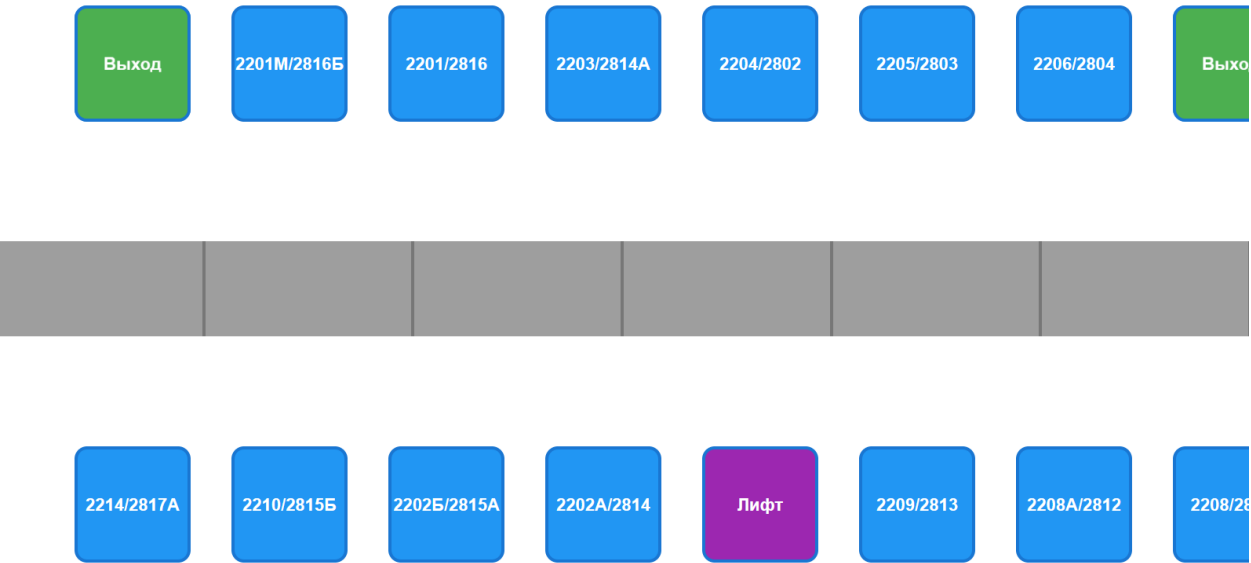


Рис.1 - Детализированный план 2 этажа с обозначением всех помещений

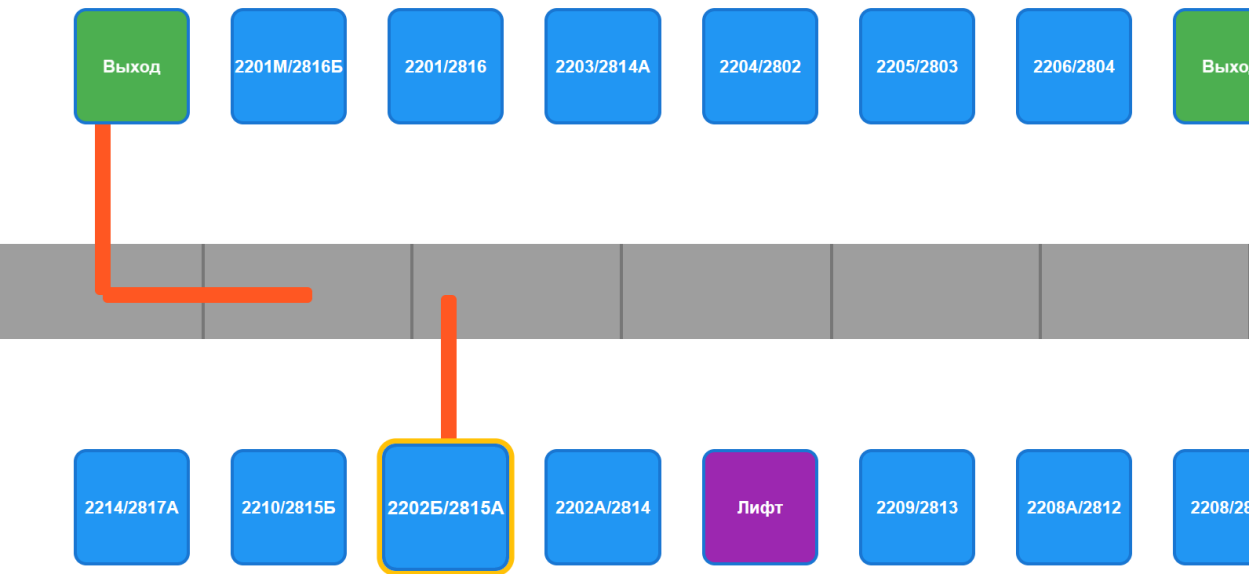


Рис.2 - Схема связей между узлами здания