

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Уральский федеральный университет  
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ  
Школа бакалавриата

## ОТЧЕТ

По проекту  
**«Разработка модуля инсоляционной горки на сайте с учетом окружающей застройки»**

по дисциплине «Проектный практикум»

Заказчик: Кузнецов Д.С.  
Куратор: Кузнецов Д.С.  
Студенты команды ChainFrost  
Борнуков Е.Е.  
Вакарчук А.А.  
Портнов Д.В.  
Темербаев К.В.

Екатеринбург, 2025

## **СОДЕРЖАНИЕ**

<b>ВВЕДЕНИЕ.....</b>	<b>3</b>
<b>Основная часть .....</b>	<b>5</b>
1.1 Вклад участников команды.....	5
1.2 Требования и backlog проекта .....	5
1.3 Анализ аналогов .....	7
1.4 Архитектура программного продукта.....	7
1.5 Методология разработки и тестирование.....	9
1.6 Планирование деятельности и распределение задач.....	9
<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>11</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....</b>	<b>12</b>
<b>ПРИЛОЖЕНИЕ А Пример JSON .....</b>	<b>13</b>

## **ВВЕДЕНИЕ**

Целью настоящего проекта является разработка программного модуля «инсоляционная горка» для интеграции на веб-сайт, предназначенного для анализа инсоляции зданий с учетом окружающей застройки. Задачи проекта включают: автоматизированную обработку геометрии зданий, расчет продолжительности прямой солнечной инсоляции в течение года, сравнение полученных данных с нормативными требованиями (в соответствии с действующими ГОСТами).

Актуальность разработки обусловлена необходимостью строгого соблюдения санитарных и строительных норм при проектировании жилых и общественных зданий в плотной городской застройке. Инструменты быстрой и точной оценки инсоляции позволяют архитектурным бюро, градостроителям и девелоперам принимать обоснованные решения уже на ранних этапах проектирования. Особенno важна такая автоматизация в условиях увеличивающейся плотности городской застройки и ограниченности свободного пространства.

Область применения программного продукта охватывает архитектурно-проектные организации, градостроительные компании, органы строительного надзора и частных застройщиков. Модуль также может использоваться в образовательных учреждениях для изучения основ инсоляционного анализа и нормативных требований к освещенности помещений.

Ожидаемым результатом реализации проекта является полнофункциональный веб-модуль, позволяющий пользователю загружать или выбирать здание на карте, рассчитывать параметры инсоляции в автоматическом режиме, получать отчет о соответствии нормативам и визуализировать «горки» инсоляции на фасадах. По завершении проекта планируется обеспечить интеграцию с существующей картографической платформой и предусмотреть возможность

расширения функциональности (например, учет прозрачных ограждающих конструкций или разной высоты оконных проемов).

## **ОСНОВНАЯ ЧАСТЬ**

### **1.1 Вклад участников команды**

Команда разработки проекта включала четырёх участников, каждый из которых внёс значительный вклад в реализацию поставленных задач:

Первый разработчик осуществлял интеграцию внешнего инструментария для расчёта инсоляции. Основное внимание было уделено использованию программного комплекса Radiance, в частности, утилитам rtrace (для выполнения расчёта освещённости в заданных точках), gensky (для генерации параметров положения солнца), oconv (для сборки сцены из компонентов). Также в обязанности входила реализация преобразования геометрических данных зданий, представленных в формате JSON(Приложение А), в формат .rad, пригодный для дальнейшей обработки средствами Radiance.

Второй разработчик занимался обработкой исходного JSON-файла, содержащего геометрию зданий. Его задачей было формирование фасадной сетки, разбиение фасадов на ячейки и подготовка данных для расчёта инсоляции по каждой ячейке. Работа велась с учётом требований к точности и устойчивости геометрических преобразований. Реализация была выполнена на языке программирования C#.

Аналитик проекта отвечал за сбор и анализ требований со стороны заказчика, а также за составление технического задания. Кроме того, аналитик занимался сопоставлением аналогичных решений на рынке и оценкой их применимости к текущей задаче.

Тимлид команды координировал взаимодействие между участниками, формировал план задач и осуществлял контроль за сроками и качеством выполнения. Также он обеспечивал техническое сопровождение проекта и вёл документацию по архитектуре и методологии разработки.

### **1.2 Требования и backlog проекта**

Сбор требований осуществлялся на основе анализа задач заказчика и потенциальных пользователей. Основная цель проекта заключалась в создании модуля, позволяющего рассчитывать инсоляцию фасадов зданий с учётом

окружающей застройки и нормативов, установленных действующим ГОСТом. Ключевые требования включали:

1. Поддержка загрузки входного JSON-файла, содержащего геометрию проектируемого здания и окружающих объектов;
2. Автоматическое построение фасадной сетки и разбиение её на ячейки;
3. Расчёт инсоляции в каждой ячейке фасада с использованием точных геометрических данных и модели солнечного освещения;
4. Учет положения солнца в зависимости от координат местности и времени года;
5. Анализ затенения существующей застройки проектируемым объектом;
6. Генерация выходных данных о соблюдении нормативов и при необходимости — корректировка формы здания;
7. Возможность визуализации результатов и экспорта данных в текстовом или графическом виде.

Для достижения этих целей был сформирован следующий backlog проекта:

1. Анализ и выбор библиотеки/инструмента для расчёта положения солнца с привязкой к геолокации. Решение: использовать утилиту gensky из пакета Radiance.
2. Разработка модуля для обработки JSON-геометрии, генерации фасадной сетки и разбиения её на ячейки. Реализация проводилась на языке C#.
3. Импорт и трансляция геометрии из JSON в .rad-файлы для использования в сценах Radiance.
4. Расчёт инсоляции для каждой ячейки с использованием rtrace и генерация отчёта о соответствии требованиям инсоляции.
5. Выявление случаев затенения окружающей застройки и, при необходимости, возврат новой геометрии с учётом корректировок.
6. Проведение модульного и интеграционного тестирования на различных этапах реализации.

7. Подготовка визуализации и документации для итоговой демонстрации проекта.

### **1.3 Анализ аналогов**

Сравнительный анализ существующих решений в области расчёта инсоляции и оценки солнечной освещённости фасадов позволил выделить несколько ключевых инструментов:

1. Autodesk Ecotect Analysis — устаревшее программное обеспечение для архитекторов и инженеров, ранее предоставлявшее возможности анализа освещённости, тепловых потоков и вентиляции. Не поддерживается и снято с распространения.
2. DIVA for Rhino — плагин для Rhinoceros, предназначенный для экологического анализа зданий. Использует движок Radiance и Daysim. Ограничен платформой Rhino, что снижает его универсальность для встраивания в сторонние веб-приложения.
3. Ladybug Tools — набор плагинов для Grasshopper и Dynamo, предоставляющий инструменты для анализа климата и инсоляции. Основан на EnergyPlus и Radiance, требует среды визуального программирования и специфического пользовательского опыта.
4. Heliodon — простое программное обеспечение для визуального анализа инсоляции, ориентировано в первую очередь на образовательные цели и не поддерживает сложные архитектурные сцены.

### **1.4 Архитектура программного продукта**

Разрабатываемый программный продукт представляет собой модуль инсоляционной горки, реализованный на языке программирования C# и взаимодействующий с внешними библиотеками для выполнения расчетов по освещенности и солнечной инсоляции. Архитектура системы основана на модульном принципе, обеспечивающем разделение функциональности и повышение гибкости разработки и сопровождения.

Основные компоненты архитектуры:

1. Модуль импорта и обработки данных — отвечает за загрузку входных JSON-файлов с геометрией зданий, их валидацию и предварительную обработку. На выходе формируются внутренние структуры, пригодные для расчетов и генерации сцены.
2. Модуль генерации сцены — на основании подготовленных данных формирует описание сцены в формате, пригодном для обработки в пакете Radiance. В рамках этого модуля выполняется преобразование геометрии в .rad-файлы, генерация источника света с использованием команды gensky и сбор сцены с помощью osconv.
3. Модуль расчета инсоляции — осуществляет вызовы к инструменту rtrace, производя расчет инсоляции по заданным параметрам. Расчеты выполняются для каждой ячейки фасадной сетки, что позволяет получить детализированную информацию о световом режиме в здании.
4. Модуль анализа результатов — анализирует выходные данные от rtrace, сопоставляет полученные значения с нормативами и определяет, соответствует ли инсоляция заданным требованиям. Также данный модуль формирует отчет о затенении по новой геометрии здания.
5. Модуль визуализации и обратной связи — отвечает за отображение результатов в пользовательском интерфейсе сайта, включая графическое представление зон инсоляции и предупреждения о несоответствии нормативам.

Компоненты взаимодействуют через четко определенные интерфейсы. Такой подход позволяет в будущем заменять или дополнять отдельные модули без необходимости переработки всей системы. Архитектура обеспечила возможность параллельной работы участников команды, что повысило эффективность процесса разработки.

Выбор архитектуры был обусловлен необходимостью интеграции с внешними инструментами (Radiance), работы с различными форматами входных данных (JSON) и обеспечением масштабируемости системы в рамках задач градостроительного анализа.

## **1.5 Методология разработки и тестирование**

Разработка программного модуля осуществлялась с применением инкрементной модели, что позволило поэтапно внедрять функционал и проводить промежуточное тестирование на каждом этапе. Такой подход обеспечил своевременное выявление и устранение ошибок, а также гибкость при внесении изменений в архитектуру и логику модуля.

В качестве инструментов управления проектом использовались система контроля версий Git и система трекинга задач Trello. Для разработки применялась среда Microsoft Visual Studio Code, обеспечивающая эффективную работу с языком C# и интеграцию с внешними библиотеками.

Промежуточное тестирование включало проверку корректности парсинга JSON-файлов, правильности генерации геометрии фасадов и сетки, а также точности расчетов инсоляции с помощью Radiance. Для верификации результатов использовались тестовые данные, разработанные аналитиком, а также сравнительный анализ с эталонными расчетами.

В процессе тестирования были выявлены ошибки, связанные с масштабированием расчетной сетки и некорректной синхронизацией координат. Эти ошибки были устранены путем оптимизации алгоритмов генерации ячеек и унификации систем координат, применяемых на разных этапах обработки данных.

Таким образом, выбранная методология разработки обеспечила качественную реализацию функционала и высокую стабильность работы программного модуля.

## **1.6 Планирование деятельности и распределение задач**

Работа над проектом велась согласно заранее составленному плану, который разбивался на две основные фазы: исследовательскую и исполнительную. Каждая фаза имела чётко определённые задачи и сроки их выполнения.

На первом этапе аналитик занимался сбором требований, анализом нормативной документации и формированием backlog проекта. Параллельно тимлид разрабатывал архитектуру модуля и организовывал процесс работы команды.

Исполнительная часть разработки была разделена между двумя разработчиками. Первый разработчик отвечал за интеграцию с инструментарием Radiance, включая создание сцен с помощью генератора солнца (gensky), конвертацию файлов и выполнение расчетов (rtrace). Второй разработчик реализовывал функционал по обработке JSON-файлов с геометрией зданий, разработке алгоритмов разбиения фасадов на ячейки и передачи данных для расчетов.

Тимлид осуществлял координацию работы, контроль сроков и качество выполнения задач.

Работа команды велась по итерационному циклу с еженедельными планёрками, на которых обсуждались результаты и корректировался план дальнейших действий. Такой подход обеспечивал гибкое реагирование на выявленные в ходе разработки проблемы и поддерживал высокую эффективность команды.

## **ЗАКЛЮЧЕНИЕ**

В ходе реализации проекта разработан программный модуль для расчёта инсоляционной горки с учётом окружающей застройки, что позволило обеспечить выполнение требований заказчика и пользователей. Полученный продукт демонстрирует способность точно анализировать инсоляционные параметры зданий в соответствии с нормативными документами и нормами ГОСТ, что подтверждается успешным прохождением промежуточных этапов тестирования.

Качество программного продукта оценивается как высокое, учитывая корректность обработки входных данных, адекватность моделирования солнечного освещения и надёжность вычислительных процедур. Тем не менее, выявленные в ходе тестирования незначительные дефекты, связанные с масштабированием расчетной сетки и обработкой граничных случаев в геометрии, не оказывают существенного влияния на общую работоспособность модуля, но требуют дальнейшей оптимизации для повышения точности и производительности.

В перспективе целесообразно внедрение дополнительных функций, таких как расширенный анализ теневых зон в динамике времени суток и года, интеграция с другими геоинформационными системами и автоматизация генерации отчетности. Также рекомендуется разработка пользовательского интерфейса для упрощения взаимодействия с модулем и расширения круга пользователей.

Таким образом, выполненный проект обеспечивает эффективное решение поставленных задач и закладывает прочную основу для дальнейшего развития программного продукта в направлении комплексного анализа инсоляции в городской среде.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. ГОСТ 12.1.005-88. Пожарная безопасность. Общие требования. – М., 1988. – 15 с.
2. Radiance — A Lighting Simulation and Rendering System. – Режим доступа: <https://radiance-online.org/> (дата обращения: 25.05.2025).
3. McNeil S., Conger M. Solar Radiation Modeling in Urban Environments // Journal of Building Performance. – 2020. – Т. 15, № 4. – С. 234–250.
4. Smith J. Introduction to JSON Processing in C# // Software Development Journal. – 2019. – Т. 12, № 2. – С. 45–53.
5. Kleeberger G. Advanced Techniques for Intersolar Calculations Using Radiance // Proceedings of the International Conference on Sustainable Building. – 2017. – С. 78–85.
6. Microsoft Docs. Working with JSON in C# – Режим доступа: <https://learn.microsoft.com/en-us/dotnet/standard/serialization/system-text-json-how-to> (дата обращения: 25.05.2025).
7. Agile Alliance. Agile Software Development – Режим доступа: <https://www.agilealliance.org/agile101/> (дата обращения: 25.05.2025).

## ПРИЛОЖЕНИЕ А

### Пример JSON

```
{  
    // Застройка пользователя  
    "buildings": [  
        {  
            "id": "building1",  
            "sections": [  
                {  
                    //Можно заменить на GeoJson  
                    "id": "section1",  
                    "coordinates": [[0, 0], [10, 0], [10, 10], [0, 10]],  
                    "floors": [  
                        {  
                            // Отрезок с 1 по 2 этаж - коммерческий с высотой 4 метра  
                            "floorStart": 1,  
                            "floorEnd": 2,  
                            "type": "commercial",  
                            "floorHeight": 4  
                        },  
                        {  
                            // Отрезок с 3 по 10 этаж - жилой с высотой 3 метра  
                            "floorStart": 3,  
                            "floorEnd": 10,  
                            "type": "residential",  
                            "floorHeight": 3  
                        }  
                    ],  
                    // Начальная высота секции от земли  
                }  
            ]  
        }  
    ]  
}
```

```
        "startHeight": 0
    }
]
}
],
// Застройка с OSM
// С OSM проблемы, см. сообщение в чате
"osm_buildings": [
{
    "id": "osm_building1",
    "coordinates": [[0, 0], [10, 0], [10, 10], [0, 10]],
    "height": 10
},
],
"parameters": {
    "window": {
        "width": 1.5,
        "depth": 0.3
    },
    "room": {
        "width": 4
    }
}
}
```