**Техническое задание на выпускную квалификационную работу**

Визуализация алгоритмов оптимизации размещения виртуальных машин в вычислительных центрах

Студент: *Митрофанов Е. Ю.*

*Р34101*

Университет ИТМО

2023 г.

# Наименования

Полное название: «3D визуализация алгоритмов размещения виртуальных машин в вычислительных центрах»

Краткое название: «Визуализация алгоритмов размещения виртуальных машин»

Условное обозначение: «Разрабатываемая визуализация»

# Назначение программного обеспечения и область применения

Оптимальное размещение виртуальных машин в центрах обработки данных - одна из ключевых проблем для поставщиков и производителей оборудования. Проблема размещения виртуальных машин является NP – сложной задачей и имеет множество разных решений, использующих различные оптимизационные алгоритмы.

Качественная и корректная визуализация алгоритмов даст наглядное представление об их работе, что поможет глубже понять происходящие процессы при выборе и сравнительном анализе этих решений, что в свою очередь способствует снижению порога входа для исследователей и инженеров в предметной области.

# Основание для разработки

* Требования к выпускным квалификационным работам. <https://edu.itmo.ru/files/345>
* ГОСТ Р 51904-2002[, **ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВСТРОЕННЫХ СИСТЕМ**](https://docs.cntd.ru/document/1200030195)
* Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, A. Beloglazov, J. A. (2012)
* Metaheuristic approaches to virtual machine placement in cloud computing: a review, Dabiah Ahmed Alboaneen, H. T. (2016).
* Energy-efficient thermal-aware autonomic management of virtualized HPC Cloud Infrastructure, Hariharasudhan Viswanathan, E. K. (2012)

# Требования к функциональным характеристикам

* Система должна предоставлять возможность выбирать конфигурацию системы, количество машин и серверов.
* Система должна предоставлять возможность выбирать конкретный алгоритм, на основе которого будет произведена симуляция
* Система должна предоставлять возможность итерационно воспроизводить работу алгоритма сопровождая ее соответствующей анимацией
* Система должна предоставлять возможность полностью симулировать полный цикл работы алгоритма для дальнейшего сравнительного анализа результата
* Система должна предоставлять возможность возврата к изначальной конфигурации для выбора другого алгоритма
* Система должна предоставлять возможность изменения выбранной конфигурации

# Требования к нефункциональным характеристикам

* Система должна собираться в исполняемое приложение на платформах Windows, а также на Web платформах под управлением WebGl
* Система должна быть понятной и графически привлекательной посредством качественного окружения, графики и качества анимации

# Структура разрабатываемой визуализации

Система должна быть реализована на графическом движке Unity Engine, иметь модульное разделение на UI, сткрипты, префабы и 3D – модели.

Приложение должно иметь возможность сборки в исполняемый exe файл для платформ Windows, а также в скомпилированный JS код для развертывания на веб-платформах под управлением WebGL.

# Пользовательский интерфейс

1. При запуске приложения пользователь конфигурирует в пользовательском интерфейсе изначальные параметры машин и серверов.
2. В пользовательском интерфейсе вбирается конкретный алгоритм, который будет использоваться при размещении.
3. Кнопка *«Next»* размещает следующий по очереди сервер на стенд в соответствии с выбранным алгоритмом.
4. Кнопка *«Solve»* размещает автоматически все объекты по стендам по порядку, без необходимости нажатия *«Next».*
5. При определенных алгоритмах пользователь при нажатии кнопки видит не перемещение одного сервера, а целую итерацию этого алгоритма – расположение всех серверов по стендам.
6. Кнопка *«Reload»* возвращает объекты в исходное положение и позволяет выбрать другой алгоритм.
7. Кнопка *«Restart»* сбрасывает установленную конфигурацию и позволяет заново выбрать и настроить ее.

# Условия эксплуатации и безопасность

Приложение может быть запущено на системах под управлением Windows в качестве исполняемого exe файла.

Приложение может быть встроено на веб-сайт как WebGL модуль, и исполнятся в браузере.

Ограничения безопасности неактуальны для разрабатываемой визуализацией, так как приложение не работает с пользовательскими данными, не общается по сети и не имеет доступа к какой-либо информации. Современный движок Unity Engine обеспечивает оптимальный просчет и рендер графики, что не приведет к потреблению большого количества вычислительных ресурсов и не будет угрожать работоспособности вычислительного устройства, на котором запускается приложение.

# Документация

Будет разработана документация трёхмерной визуализации и процесса разработки с перечислением используемых инструментов и анализом предметной области. Документация должна быть разработана в соответствии с ГОСТ Р ИСО/МЭК 15910-2002 [ПРОЦЕСС СОЗДАНИЯ ДОКУМЕНТАЦИИ ПОЛЬЗОВАТЕЛЯ ПРОГРАММНОГО СРЕДСТВА](https://docs.cntd.ru/document/1200030141)

# Стадии и этап разработки

1. Исследование предметной области
2. Анализ применимых инструментов для создания визуализации
3. Разработка прототипа приложения
4. Тестирование и анализ прототипа
5. Разработка полноценного приложения на графическом движке
6. Сборка и размещение приложения на веб – платформе
7. Тестирование разработанного приложения для дальнейшего анализа результатов тестирования
8. Написание документации к разработанному приложению

# Порядок контроля и приёма

После каждого этапа разработки осуществляется анализ и, если возможно, тестирование полученных решений. Соответствующие выводы должны содержаться в ВКР.

Финальными этапами является проверка ВКР на антиплагиат и защита.