

Тестовое задание для C# Unity Network Developer

Контекст

Цель:

- Собрать простой мультиплеерный прототип для оценки навыков сетевого кода и качества реализации.

Сроки и оплата

- Базовый минимум: 16 ч
- Дополнительные задачи: 8 ч
- Время на выполнение - 7 рабочих дней начиная со следующего дня, после получения ТЗ
- Оплата за выполнение ТЗ - 2000р. При условии выполнения всех пунктов из основных задач.

Окружение

- Unity: 6000.0.58f2
- Сетевая библиотека: любая по вашему выбору
(предпочтительно [FishNet](#), выбор не влияет на оценку)
 - Кратко обосновать выбор в [README.md](#)

Описание задачи

- 3D-сцена с ограниченной ареной (размер на ваше усмотрение, небольшая как в *Brotato*)
- Вид сверху (top-down), камера закреплена на локальном игроке

- **Движение игрока** в одной плоскости (например, XZ):
 - Игрок не выходит за пределы арены
 - Передвижение на **WASD**
- **Игрок** смотрит в сторону ближайшего противника
 - При его отсутствии — по направлению движения
- **Автоматическая атака:** проджектайл по кулдауну
 - Направление — взгляд игрока
 - Попадание наносит урон первому врагу по пути
 - Враг умирает после нескольких попаданий
- **Два типа врагов** — различимы визуально, с разной скоростью движения
- **Поведение врагов:**
 - При **спавне** выбирают ближайшего игрока
 - Постоянно движутся к нему
 - Наносят урон при контакте
- **Спавн врагов:** непрерывный, с интервалом
 - Можно ограничить максимальное число активных врагов
- **Получение урона игроком** сопровождается визуальным эффектом

UI

- Счётчик активных врагов
- Пинг
- Пропускная способность сети

Сетевая часть (минимум)

- Синхронизация положения игроков
- Спавн и попадание проджектайлов
- Движение противников
- Синхронизация получения урона (игрок / противник)
- Синхронизация счётчика врагов

Дополнительные задачи

- Кнопка для **спавна дополнительных юнитов**

- Второй тип проджектайла — **самонаводящийся на цель**
- Использование **батчинга** или **роллбека** для синхронизации позиций противников
- Возможность **увеличить задержку (пинг)**
- Описать в **README.md** как бы вы **улучшили/изменили** получившийся прототип, будь на него больше времени

Примечания

- Допустим **хардкод параметров** (скорость, урон, кулдауны и т.д.)
- Реализация **смерти игрока** — необязательна
- Допускаются оба подхода:
 - Сервер-авторитарный
 - Клиент-авторитарный

Сдача тестового задания

Что сдавать

- Ссылка на репозиторий **Git** с проектом
- Файл **README.md**, содержащий:
 - Как запустить (Host/Client), какие порты/адреса используются
 - Какой выбран сетевой подход и почему
 - (Опционально) описание других решений и их причины
- (Опционально) **билд Windows x64**

Что оценивается

- Наполненность и ясность **README.md**
 - Обоснование подходов и решений
- **Качество синхронизации** объектов при росте их количества
- Влияние **высокого пинга** на игровой процесс и сетевые задержки