

МНОГОПОЛЬЗОВАТЕЛЬСКАЯ КООПЕРАТИВНАЯ ИГРА С СОЦИАЛЬНЫМ ВЗАИМОДЕЙСТВИЕМ

Многопользовательская игра является режимом компьютерной игры, во время которого играет больше одного человека. Кооперативная игра – это тип многопользовательской игры, в которой игроков объединяет общая цель, выполнению поставленной цели препятствуют компьютерные соперники. Для достижения многопользовательской составляющей существует подходы такие, как игра на одном экране, поочерёдная смена игроков, система из одного компьютера и нескольких терминалов, игра по сети.

Система мультиплеера состоит из серверного и клиентского модулей.

Основной сервер написан на ASP.NETCore[1] и сможет удаленно разворачивать клиентские модули в режиме сервера для игры по интернету. На серверной стороне для хранения данных о пользователях используется система управления базами данных SqlLite. Целью серверного модуля является поддержка функционала, связанного с социальным взаимодействием, таким как функции обмена сообщениями в реальном времени в глобальном чате, функции обмена личными сообщениями от пользователя пользователю, добавление в список друзей(одностороннее), удаление из списка друзей, отображение актуального статуса пользователя, система аутентификации и регистрации новых пользователей. Основной задачей серверного модуля многопользовательской кооперативной игры с социальным взаимодействием является запуск и управление серверами игры, которые позволяют пользователям подключиться к нему и обмениваться информацией по модели клиент-сервер.

Система нотификаций в реальном времени позволяет уведомить пользователя о каком-либо событии для обработки.

Клиентский модуль представлен приложением на Unity[2], который обменивается сообщениями с сервером посредством SignalR[3] и HTTP запросов.

Для сетевых игровых сессий клиентский модуль должен иметь возможность использования в режиме сервера. В данном режиме в клиентском модуле отключается графическое отображение объектов, при запуске модуль переходит в режим хоста и начинает прослуши-

вать указанный порт для дальнейшего подключения игроков. Когда в режиме сервера не остается игроков, модуль должен отправить основному серверу запрос на завершение процесса модуля.

Сетевые игровые сессии осуществляются с помощью UNET и основного сервера. Игрок, создающий игровую сессию, отправляет серверу запрос на создание игрового сервера и рассылку уведомлений приглашенным игрокам. Далее при принятии приглашения игроки подключаются к выделенному игровому серверу.

ЛИТЕРАТУРА

1. Asp.NetCore Полное руководство [Электронный ресурс]. – Режим доступа: <https://metanit.com/sharp/aspnet5>. Дата доступа: 18.04.2020
2. Unity (игровой движок) [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/Unity_\(игровой_движок\)](https://ru.wikipedia.org/wiki/Unity_(игровой_движок)). Дата доступа: 18.04.2020
3. Что такое SignalR? [Электронный ресурс]. – Режим доступа: http://codehint.ru/articles/2013-08-16_introduction_signalr. Дата доступа: 18.04.2020

УДК 557.114:616-006

Студ. Д. И. Новицкий
Науч. рук. ассист. Н.А. Северинчик
(кафедра программной инженерии, БГТУ)

МУЗЫКАЛЬНЫЙ СТРИМИНГОВЫЙ СЕРВИС

Стриминговые (они же потоковые) сервисы работают по принципу передачи контента от провайдера к пользователю. Весь контент уже загружен на стороннем сервере, конченому пользователю не требуется ничего скачивать для просмотра или прослушивания. Контент транслируется в режиме реального времени, скорость подгрузки напрямую зависит от скорости интернета пользователя. С нынешним даже самым простым интернетом можно без проблем прослушивать музыку и просматривать видео со стриминговых сервисов.

Клиентская часть музыкального стримингового сервиса будет реализована с использованием платформы для разработки веб-приложений Angular[1]. Он поддерживается на разных платформах (веб, мобильные устройства, нативный десктоп), он мощный, современный и у него отличная экосистема.

Серверная часть будет реализована с использованием технологий ASP.NETCore[2] и EntityFrameworkCore. Для хранения данных о