

уже готовых исходных модулей помогает оптимизировать работу по созданию игр, упрощать осуществление рутинных операций, освобождая место для творческих задач, превращая программирование из ремесла в искусство. Разработке некоторых элементов такого модуля посвящена настоящая работа. Базовые компоненты разработки игровых приложений были подготовлены вначале для упрощенного игрового интеллекта (ИИ), который умеет двигаться к цели и атаковать врага с определенной задержкой, а затем использованы с некоторой доработкой для продвинутого ИИ, который также умеет двигаться к цели и атаковать врага с определенной задержкой, а также – преследовать цель, потерянную из зоны видимости, обходить и разрушать препятствия.

Алгоритм работы упрощенного ИИ реализован в классе AI.js, его основные атрибуты выполняют следующие функции: target – цель, к которой движется ИИ; moveSpeed – скорость движения к цели; rotationSpeed – скорость поворота к цели; maxDistance – максимальная удаленность от цели; curDistance – текущая удаленность от последней цели; reacDistance – дистанция реагирования на цель; attackSpeed – скорость атаки ИИ; attackcheck – флаг задержки между атаками. Методы класса AI.js реализуют следующие действия: Start() – инициализация ИИ; Update() – проверка наличия врагов в зоне видимости, движение игрока к цели; Attack() – атака игрока; Reload() – таймер задержки перед следующей атакой.

Алгоритм работы продвинутого ИИ реализован в классе AIImproved.js, его основные атрибуты выполняют следующие функции: state\_stand – состояние бездействия; state\_walk – состояние движения; state\_attack – состояние атаки; condition\_see\_enemy – условие видимости врага; condition\_can\_attack – условие атаки врага, condition\_can\_stand – условие бездействия, condition\_can\_walk – условие движения; condition\_can\_obstacle – условие наличия препятствия; update() – обдумывание действий, выбор обстановки, получение текущего набора действий; selectNewSchedule(condition:Array) – исполнение текущего набора действий; getCondition() – генерация и возврат новых условий; onInitStand() – установление ожидания нового действия; onStand() – переключение на новое действие; onInitWalk() – установка решений по движению; onWalk() - движение.

Базовые компоненты ИИ разработаны на основе игровой среды Unity3D и языка C#. Они могут использоваться начинающими программистами для изучения процесса и создания своих игр.

**Е. В. Лукашкин**

*Науч. рук. В. А. Ковалёва,*

*канд. физ.-мат. наук, ст. преподаватель*

## **ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС ПО АВТОМАТИЗИРОВАННОЙ ОТПРАВКЕ SMS-СООБЩЕНИЙ ПОСРЕДСТВОМ ТЕХНОЛОГИИ GSM**

Одним из способов автоматизации процесса информирования заинтересованных лиц и рассылки информационных уведомлений является обеспечение связи через мобильные технологии путем использования соответствующих устройств и стандартов. Представителем таких стандартов может быть GSM-стандарт, одной из составных частей которого является служба коротких сообщений (SMS), повышающая не только уровень общения, но и степень информирования.

Нами была разработана основа программно-аппаратного комплекса по автоматизированной отправке SMS-сообщений. Были исследованы аспекты организации отправки и приема SMS-сообщений, проведен анализ требований, предъявляемых

к системам и комплексам подобного рода. Для решения задачи автоматизированной отправки SMS-сообщений была составлена и реализована информационная модель, а также выбраны средства разработки, которые позволят наиболее эффективно разрабатывать программную часть системы. Для создания комплекса была проработана аппаратная часть, в который были выбраны подходящие устройства для реализации комплекса.

Отметим, что одной из сфер применения программно-аппаратного комплекса является работа учебных учреждений. Разрабатываемая система обеспечит высокий уровень информирования о текущей и семестровой информации, что позволит повысить эффективность работы администрации учебного заведения с родителями студентов и прочими заинтересованными лицами. Это в свою очередь качественно скажется на успеваемости и уменьшит количество пропусков студентами, а также даст возможность экономии денежных средств и времени для организации такого информирования другим образом.

*Е. Ю. Лукашов-Фурсиков*  
*Науч. рук. В. А. Короткевич,*  
*канд. техн. наук, доцент*

## **АВТОМАТИЗАЦИЯ ОБМЕНА ИНФОРМАЦИЕЙ МЕЖДУ СЕРВЕРАМИ БАЗ ДАННЫХ ПРОГРАММНОГО КОМПЛЕКСА «ПРИЕМНАЯ КОМИССИЯ ВУЗА»**

Программный комплекс «Приемная комиссия вуза» использует два SQL-сервера. Первый используется оконными приложениями, которые предназначены для секретарей приемной комиссии и руководства вуза. На втором хранятся данные, используемые интернет-сервисом для абитуриентов – «Личный кабинет». Разделение данных на два сервера вызвано требованиями производительности, надежности и защищенности системы.

В процессе функционирования программного комплекса обеспечивается обмен данными между серверами. При этом анкетные данные, введенные абитуриентами в личном кабинете, копируются на сервер, доступный секретарям приемной комиссии, а сообщения для абитуриентов, данные таблиц-справочников, данные текущего конкурса и проходные баллы копируются на сервер интернет-сервиса для отображения абитуриентам.

Для выполнения поставленной задачи в среде C++ Builder была разработана программа с многостраничным интерфейсом. В программе реализовано два способа копирования данных: по запросу администратора программного комплекса и автоматически (по расписанию).

Работа по расписанию может выполняться либо по периоду, либо по времени. Для работы по периоду с помощью полей ввода задается временной интервал с точностью до минут, а также количество минут, через которое будет производиться очередное копирование данных. Предусмотрена возможность круглосуточного запуска. При работе по времени необходимо задать список времен с точностью до минут, согласно которому будет выполняться копирование данных. При необходимости данный список можно редактировать. Независимо от временной конфигурации помечаются дни недели, в которые будет выполняться обмен данными. Реализована возможность в случае необходимости задавать конечную дату и время, при достижении которой процесс копирования данных прекращается.

Для копирования проходных баллов и текущего конкурса дополнительно реализованы компоненты интерфейса, отвечающие за конфигурацию копирования каждой