**Научно-исследовательская часть  
  
2.1. Постановка задачи проектирования**Задача курсовой работы состоит в разработке децентрализованного peer-to-peer чата на основе Windows API, где сообщения передаются с одного компьютера на другой.

**2.2. Описание предметной области**

Чат – средство обмена сообщениями в реальном времени с помощью компьютера, смартфона, планшета в основном через интернет. Чат появился в результате развития информационных технологий. Ранней альтернативой чату был телефон, который позволял общаться в режиме реального времени.

Существует несколько разновидностей программной реализации чатов:

* HTTP или веб-чаты. Такой чат выглядит как обычная веб-страница, где можно прочесть последние несколько десятков фраз, написанные участниками чата и модераторами. Страница чата автоматически обновляется с заданной периодичностью.
* Чаты, использующие технологию Adobe Flash. Вместо периодической перезагрузки страницы между клиентом и сервером открывается сокет, что позволяет моментально отправлять или получать сообщения, расходуя меньше трафика.
* IRC, специализированный протокол для чатов.
* Программы-чаты для общения в локальных сетях (например, Vypress Chat, Intranet Chat, Pichat). Часто есть возможность передачи файлов.
* Чаты, реализованные поверх сторонних протоколов (например, чат, использующий ICQ).
* Чаты, работающие по схеме клиент-сервер, это позволяет использовать их в сетях со сложной конфигурацией, а также управлять клиентскими приложениями (например, Mychat, Jabber)
* Чаты, работающие в одноранговых сетях. У них нет потребности в отдельном сервере, они часто используют возможности технологий DHT и TCP Relay (пример: Tox)
* Чаты, использующие технологию Push. Вместо периодической отправки запросов серверу о новых сообщениях, используются входящие сообщения от сервера, что позволяет отправлять и получать сообщения, расходуя меньше трафика (например, WinGeoChat)
* Полностью анонимные чаты. В них собеседник не предполагает с кем общается и при каждом новом соединении общается с новым человеком. (например, ChatVdvoem (ЧатВдвоем))

Идея peer-to-peer общения заключается в том, что каждый peer знает и поддерживает информацию о других участниках. Когда новый клиент подключается к сети, он может узнать у любого пира информацию о том, где и какие файлы сейчас доступны. Когда клиент начинает скачивает файл себе на компьютер, то скачанные части этот файла сразу становятся доступны для скачивания другим пользователям. Никто не даёт гарантию, что каждый клиент будет находиться длительное время в сети и давать скачивать информацию, напротив -- ситуация, когда клиент пропадает в процессе загрузки, является естественной. В данном случае будет найден новый клиент, который может продолжить передачу данных. Для поддержания списка активных peer-ов каждый клиент посылает другим клиентам heartbeat. Heartbeat - это сообщение, которое один сервер посылает другому, чтобы сказать ему, что он жив. Соответственно, если heartbeat долго не приходит, значит этот сервер нужно удалить из списка активных peer-ов. Постоянно обмениваться heartbeat-ом с большим количеством серверов трудоёмко. Поэтому у каждого сервера есть два параметра - нижняя и верхняя граница на размер списка активных клиент. Когда это количество становится ниже нижней границы, запускается поиск новых участников. клиент запрашивает у других клиент список активных peer-ов.

**2.3. Анализ аналогов и прототипов**

**RetroShare** – известное, свободное кроссплатформенное программное обеспечение для бессерверного обмена письмами, мгновенными сообщениями, построенное на P2P.

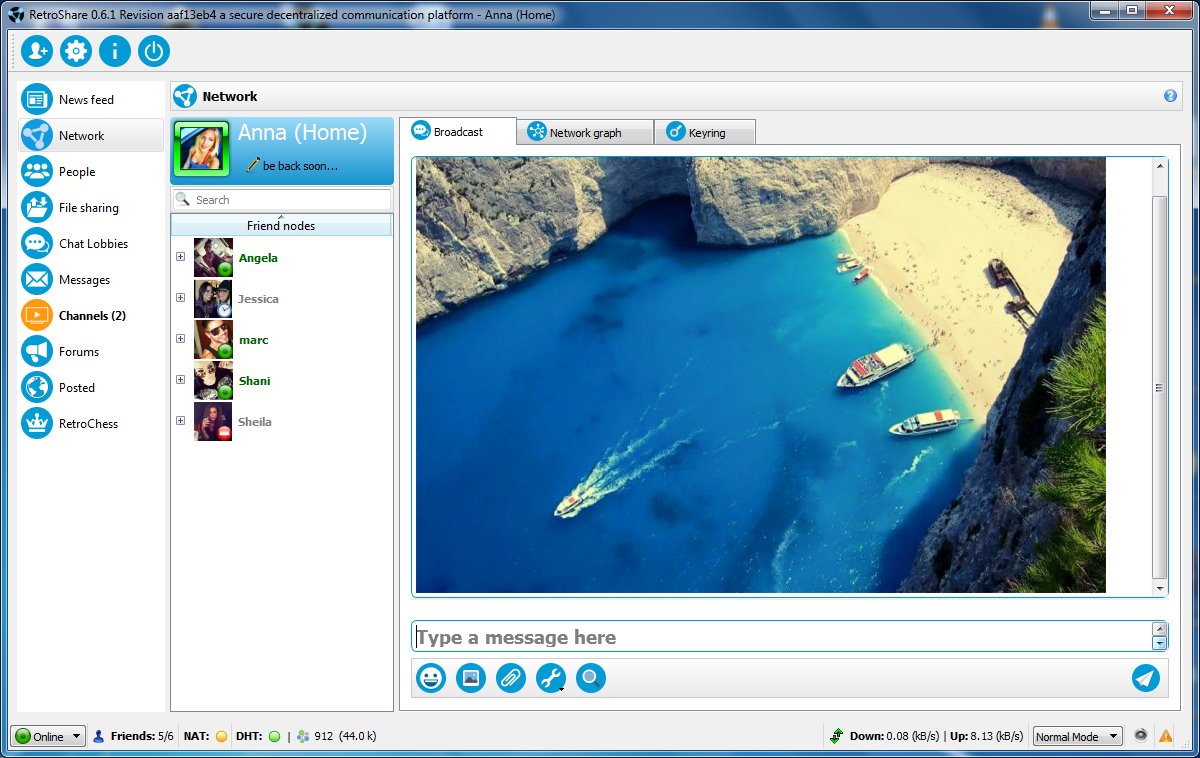


Рисунок.1 Интерфейс программы «RetroShare».

Система позволяет организовывать обособленные бессерверные Friend-to-Friend сети или одну глобальную сеть с определённым кругом лиц — например, друзьями, семьёй, коллегами. Соединения устанавливаются напрямую только с теми участниками, которым дали разрешение. Это важная особенность, перемещающая подобные сети в рамки особого класса децентрализованных сетей, в которых несанкционированное подключение к вашему компьютеру запрещено.

Обмен сообщениями возможен от имени анонимных личностей, так и непосредственно между сетевыми узлами. В первом случае доступна оффлайн доставка, применяется хранение сообщений на узлах общих друзей.

Система анонимных личностей также предоставляет возможность общения с людьми за пределами круга друзей, сообщения передаются по цепочке из последовательно соединенных узлов. Таким образом возможна связь с любым участником сети, в том числе с тем, с которым соединение не установлено.

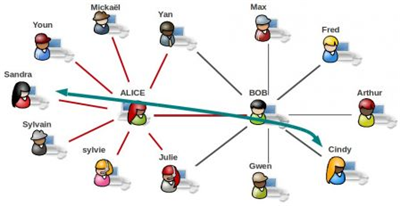
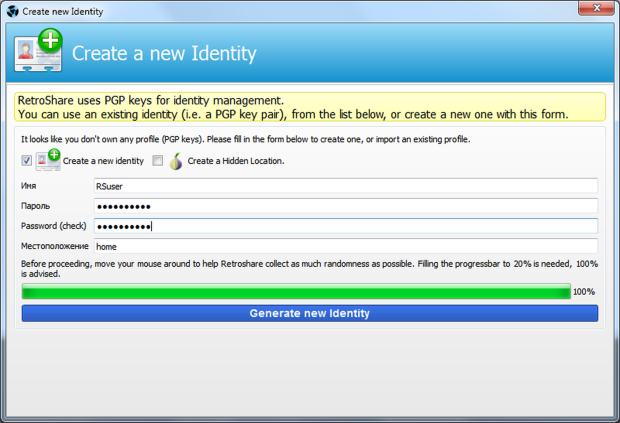


Рисунок.2 Пример общения людей за пределом круга друзей.

Процедура подключения к сети RetroShare включает в себя два этапа: создание личного сертификата и поиск доверенных пиров.

При первом запуске на экран выводится окно, помогающее пользователю создать личный сертификат.

Рисунок.3 Создание личного сертификата.



Доверенных пиров можно найти в чате внутри. К чату можно подключиться, обменявшись сертификатом с роботом. Следует отметить, что взаимный обмен сертификатами с одним или обоими роботами лишь позволяет быстро загрузить основные публичные чаты и не даёт возможности пользоваться ресурсами сети. Это значит, что вы не сможете производить поиск и просмотр контента. Для этого пользователь должен найти как минимум одного участника сети, готового осуществить взаимный обмен сертификатами. Чат полезен в случае, если требуется наладить временный приватный контакт с человеком или произвести обмен сертификатами, когда передача сертификатов через общие чат-комнаты нежелательна. Удалённый контакт реализуется через систему анонимных туннелей и возможен лишь в том случае, когда оба участника уже вошли в сеть.

Для того, чтобы начать общение необходимо, чтобы кто-то откликнулся на просьбу и дал ссылку на собственный сертификат. Пользователь добавляет доверенного пира, а также вводите пароль к сертификату, созданному ранее.

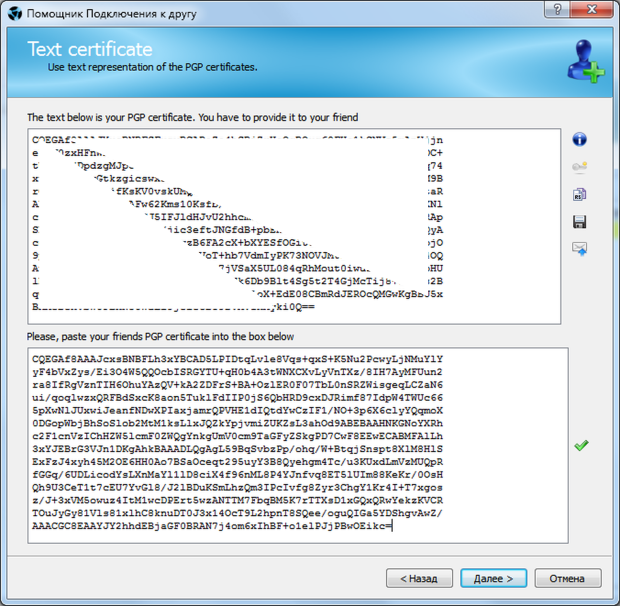


Рисунок.4 Для подключения к роботу помещаем его сертификат в нижнее окно.

Появляется окно прогресса подключения. Можно нажать «OK», и подключение к пиру будет проходить в фоновом режиме.

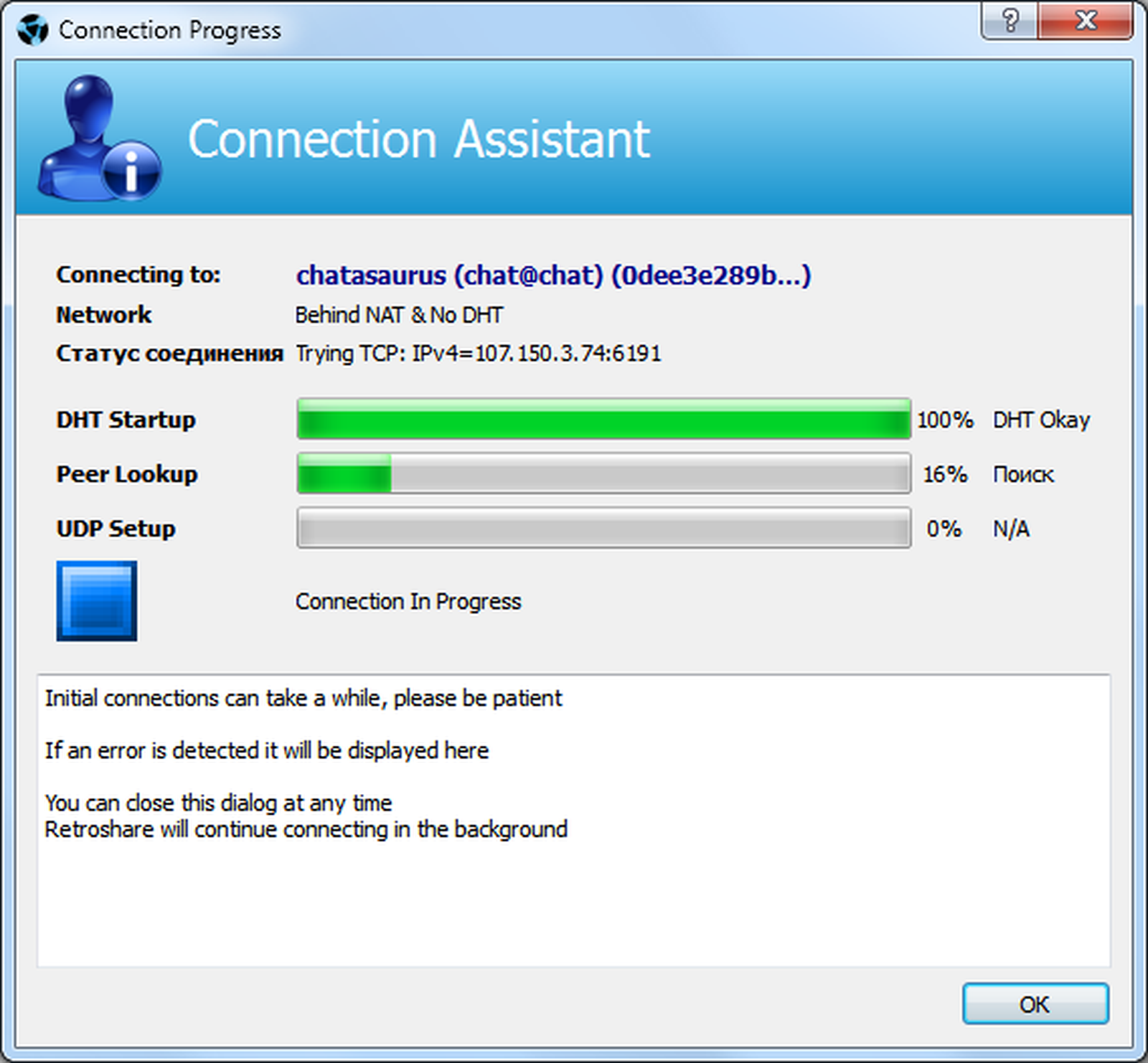


Рисунок.5 Процесс подключения клиента к роботу.

После этого, во вкладке друзья отображается полноценный доверенный участник. При подключении хотя бы к одному действующему участнику сети, пользователь также автоматически становится действующим участником.

Публичные чаты не модерируются и не цензурируются. Это значит, что там отсутствует администрация в той или иной форме, а также возможность наложения запрета на доступ к чат-комнате — бана. Для защиты от спама имеется возможность на уровне клиента установить так называемый «немой режим» (mute) для нарушителя. В результате клиент будет игнорировать сообщения от одного или нескольких пользователей чата, занимающихся рассылкой спама.

**MyChat -** клиент-серверный чат, идеально подходит для передачи сообщений и обмена файлами в офисе или компании, может работать внутри корпоративной сети, а также через Интернет. Будет полезен компаниям, которым нужно средство корпоративного общения, где недопустимо использование таких публичных IM систем, как ICQ или Skype, в связи с ограниченным доступом к Интернету или корпоративными требованиями к безопасности для предотвращения утечки информации.

Клиент позволяет подключаться к отдельным каналам, производить регистрацию в каналах, вести приватные разговоры, передавать файлы, просматривать доски объявлений, искать пользователей, создавать оповещения с различными параметрами.

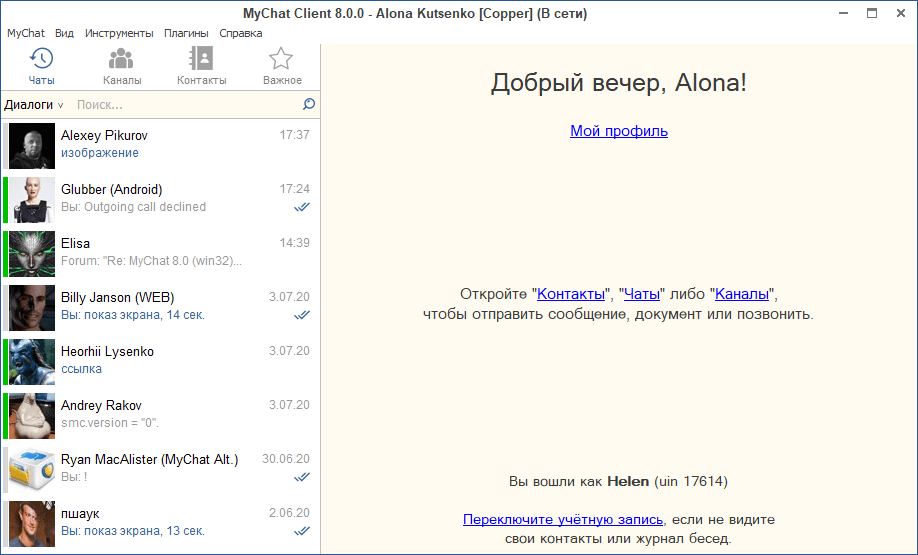


Рисунок.6 Интерфейс программы «MyChat».

В силу того, что в оповещениях нельзя вести диалог, как в обычном приватном разговоре, существует отправка многоадресных приватных сообщений. Выделяете несколько людей в списке контактов мышкой с зажатой клавишей Ctrl или Shift и отправляете приватное сообщение:

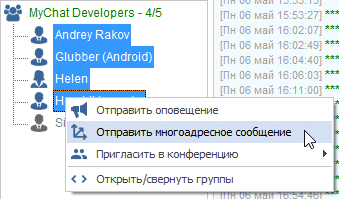


Рисунок.7 Процесс отправления многоадресных сообщений.

Все отправленные сообщения попадают к в историю, но диалоги не открываются, чтобы не плодить лишние окошки. Если придёт ответ, программа сразу это покажет.

Для частых однотипных отправок можно пользоваться собственными уже готовыми списками людей, как в оповещениях:

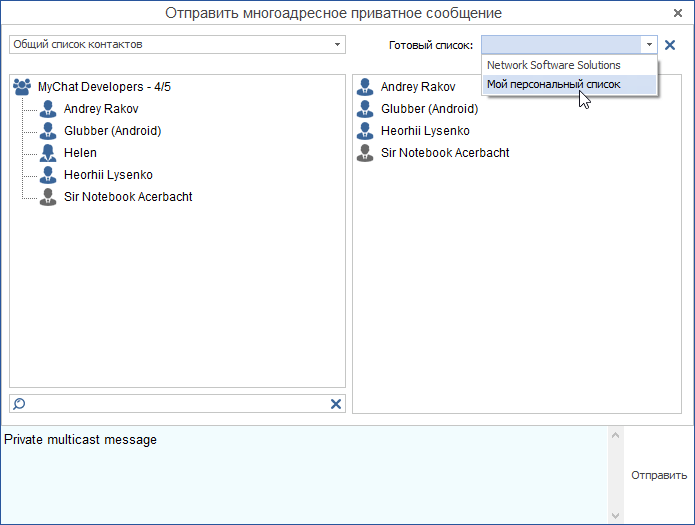


Рисунок.8 Списки контактов клиента.

**2.4. Перечень задач подлежащих решению в процессе разработки**

1. Должны быть реализованы стандартные функции работы чата: задание псевдонима,ввод портов для подключения, отправка и получение сообщений.

2. Программа должна позволять пользователю менять ряд настроек. В частности, имя пользователя в чате, шрифт и размер шрифта текстового поля, а также должна быть реализована возможность сохранения логфайлов.

3. Программный продукт должен иметь дружественный интерфейс, не требующий специальной подготовки пользователей, должен способствовать более эффективному распределению и экономии времени.

**2.5. Обоснование выбора инструментов и платформы для разработки**

В операционной системе Windows реализована объектноориентированная идеология. Базовый объект системы – окно, поведение которого определяется методом, называемым функцией окна. Графический образ окна на экране дисплея – прямоугольная рабочая область. Независимо от своего типа любой объект Windows идентифицируется описателем или дескриптором (handle). Все взаимоотношения программного кода с объектом осуществляются только через его дескриптор. Интерфейс прикладного программирования (API – Application Programming Interface) представляет собой совокупность 32-битных функций (Win32 API), которые предназначены для создания приложений (программ), работающих под управлением Microsoft Windows. Функции объявлены в заголовочных файлах. Главный из них − файл windows.h, в котором содержатся ссылки на другие заголовочные файлы.

API — это аббревиатура названия Application Programming Interface (интерфейс прикладного программирования). API представляет собой совокупность функций и инструментов, позволяющих программисту создавать приложения (программы), работающие в некоторой среде.

Win32 API — это набор функций для создания программ, работающих под управлением Microsoft Windows 98, Windows NT или Windows 2000. Все функции этого набора являются 32 битными, что отражено в названии интерфейса.

Для выполнения курсовой работы был выбран язык программирования C++.

C++ — компилируемый, статически типизированный язык программирования общего назначения.

Поддерживает такие парадигмы программирования, как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование. Язык имеет богатую стандартную библиотеку, которая включает в себя распространённые контейнеры и алгоритмы, ввод-вывод, регулярные выражения, поддержку многопоточности и другие возможности. C++ сочетает свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником — языком C — наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования.

C++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его

применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также игр.

Синтаксис C++ унаследован от языка C. Одним из принципов разработки было сохранение совместимости с C. Тем не менее C++ не является в строгом смысле надмножеством C; множество программ, которые могут одинаково успешно транслироваться как компиляторами C, так и компиляторами C++, довольно велико, но не включает все возможные программы на C.

Одними из отличительных особенностей и очень важных преимуществ C++ является:

Поддерживаются различные стили и технологии программирования, включая традиционное директивное программирование, ООП, обобщённое программирование, метапрограммирование (шаблоны, макросы).

Предсказуемое выполнение программ является важным достоинством для построения систем реального времени. Весь код, неявно генерируемый компилятором для реализации языковых возможностей (например, при преобразовании переменной к другому типу), определён в стандарте. Также строго определены места программы, в которых этот код выполняется. Это даёт возможность замерять или рассчитывать время реакции программы на внешнее событие.

Автоматический вызов деструкторов объектов при их уничтожении, причём в порядке, обратном вызову конструкторов. Это упрощает (достаточно объявить переменную) и делает более надёжным освобождение ресурсов (память, файлы, семафоры и т. п.).

Пользовательские функции-операторы позволяют кратко и ёмко записывать выражения над пользовательскими типами в естественной алгебраической форме.