|  |  |
| --- | --- |
| *voenmeh* | МИНОБРНАУКИ РОССИИ  федеральное государственное бюджетное образовательное учреждение  высшего образования  **«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**  **(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова»)** |
| БГТУ.СМК-Ф-4.2-К5-01 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Факультет |  | И |  | Информационные и управляющие системы |
|  |  | шифр |  | наименование |
| Кафедра |  | И9 |  | Систем управления и компьютерных технологий |
|  |  | шифр |  | наименование |
| Дисциплина |  | научно-исследовательская работа в семестре | | |

КУРСОВАЯ РАБОТА

на тему

|  |
| --- |
|  |
| Выбор функций, структуры и средств |
| проектирования для электронной обучающей системы |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил студент группы | | | |  | И9М31 |
| Разомазов А.В. | | | | | |
| Фамилия И.О. | | | | | |
| **РУКОВОДИТЕЛЬ** | | | | | |
| Черкасов О.Ф. | |  |  | | |
| Фамилия И.О. Подпись | | | | | |
| Оценка |  | | | |  |
| «\_\_\_\_\_» |  | | | | 201\_г. |

САНКТ-ПЕТЕРБУРГ

2018 г.

Содержание

[ВВЕДЕНИЕ. 3](#_Toc524032267)

[ПОСТАНОВКА ЗАДАЧИ 5](#_Toc524032268)

[1 ОСНОВНОЙ НЕОБХОДИМЫЙ ФУНКЦИОНАЛ ОС 6](#_Toc524032269)

[2 ТИП И СТРУКТУРА ОС 10](#_Toc524032270)

[2.1 ВЫБОР ПРИНЦИПА ПОСТРОЕНИЯ ОС 11](#_Toc524032271)

[2.2 ВЫБОР СТРУКТУРЫ ОС 15](#_Toc524032272)

[3 ОБОСНОВАНИЯ ВЫБРАННОГО ЯЗЫКА ПРОГРАММИРОВАНИЯ 18](#_Toc524032273)

[ЗАКЛЮЧЕНИЕ 22](#_Toc524032274)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 24](#_Toc524032275)

**Введение.**

Автоматизированные обучающие системы представляют собой программно-технический комплекс, включающие в себя методическую, учебную и организационную поддержку процесса обучения, проводимого на базе информационных технологий. Автоматизированные обучающие системы своим появлением способствовали расширению наших представлений о возможностях электронно-вычислительных машин (ЭВМ) прежде всего потому, что открыли новые перспективы в деле совершенствования образования.

Применение ЭВМ в системе образования явилось следствием появления новых направлений как в использовании технических средств обучения, так и в методах и формах самого процесса обучения. В настоящее время обучающие системы используются во многих образовательных учреждениях.

Рассматривая электронные обучающие системы (ЭОС) как средства автоматизации обучения, повышения качества образования, а также возможность получения знаний без контакта обучающегося и преподавателя, стоит сразу же отметить: универсальных систем обучения не существует. Проблема заключается в том, что каждый обучаемый человек индивидуален, имеет значение специфика направления учебного курса и сам принцип обучения. Проектируя модель ЭОС необходимо сразу определить задачи, такие как:

* уяснение задачи, в результате чего определяется проблемная ситуация, устанавливается структура данных и неизвестных;
* разработка математической модели ситуации;
* составление плана решения (в виде текста, блок-схемы, графа и т.п.);
* выбор (разработка) алгоритма, наиболее эффективного для применения компьютера;
* составление программы в соответствии с алгоритмом;
* отладка и тестирование программы;
* проверка пригодности модели;
* оценка эффективности принятого плана решения;
* организация взаимодействия с компьютером.

При разработке ЭОС требуется сразу определить цели, которые данная система будет решать в рамках образовательного процесса. Важно отметить, что лучшие обучающие системы не насыщенны сложными данными, формулами, схемами, совмещены с основным обучением, а сам процесс нередко дополняют забавные фразы, шутки, игры. Анализ и практика показывают, что положительные эффекты обучения наиболее отчетливо проявляются при:

* изучении базиса дисциплины, ее сложных закономерностей и алгоритмов, динамических процессов;
* реализации игр и имитаций;
* организации исследовательских и тренирующих процессов;
* автоматизации самоконтроля, контроля, оценки обучения;

Стоит отметить, что существуют различные, по типу работы и поставленной задаче, системы обучения. Все эти критерии показывают, что разработка ЭОС включает в себя не только технические методы реализации, выбор типа и структуры системы, но и методы предъявления учебной информации. Усложняется разработка тем, что стандартов на построение подобных систем нет.

Целью данной работы является разработка обучающей программы по проектированию по дисциплине “Проектирование в CAD и CAE системах, работа с электронным документооборотом”.

# **Постановка задачи**

Для реализации поставленной цели были определены основные задачи:

* Выделить основной функционал обучающей системы (ОС)
* Выбрать тип и структуру ОС
* Определить программные средства для реализации.

Разработать упрощенную обучающую программу создать простой интерфейс для пользователя без потери функционала программного обеспечения (ПО) и создать простой редактор, благодаря которому преподаватель сможет сформировать обучающий курс и, по необходимости, добавить тесты или проверочные задания для оценки полученных знаний.

# **1 Основной необходимый функционал ОС**

Необходимость в электронной системе обучения появилась из-за появления современных методов проектирования и проверки изделия, которые тяжело даются специалистам, не работавших раньше с актуальным программным обеспечением.

Преимущества использования компьютера в обучении:

* адаптивность учебного материала;
* многотерминальность (одновременная работа группы пользователей);
* интерактивность (взаимодействие ТС и учащегося, имитирующее естественное общение);
* подконтрольность индивидуальной работы обучаемых во внеучебное времяДовольно распространенная проблема проектирования – незнание или непонимание специалистов принципов работы современных программных комплексов для проектирования.

Предполагается, что разработка будет внедряться совместно с традиционным процессом обучения. Обучающая система будет предполагает наличие обратной связи в триаде «Педагог – Обучающая программа – Обучаемый» (рисунок 1.1).

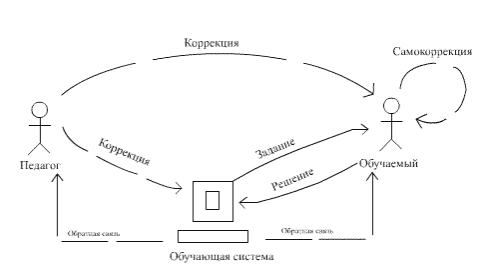
****

Рис. 1. Схема взаимодействия объектов«Педагог – Обучающая программа – Обучаемый»

Основным ПО, с которым работают инженеры являются:

* **SolidWorks** - программный комплекс САПР для автоматизации работ промышленного предприятия на этапах конструкторской и технологической подготовки производства. Обеспечивает разработку изделий любой степени сложности и назначения
* **AutoCAD** - двух- и трёхмерная система автоматизированного проектирования и черчения, разработанная компанией Autodesk. Первая версия системы была выпущена в 1982 году. AutoCAD и специализированные приложения на его основе нашли широкое применение в машиностроении, строительстве, архитектуре и других отраслях промышленности.
* **Altium Designer** - комплексная система автоматизированного проектирования (САПР) радиоэлектронных средств, разработанная австралийской компанией Altium. Ранее эта же фирма разрабатывала САПР P-CAD, который приобрёл необычайную популярность среди российских разработчиков электроники.
* **ANSYS** — универсальная программная система конечно-элементного (МКЭ) анализа, существующая и развивающаяся на протяжении последних 30 лет, является довольно популярной у специалистов в сфере автоматизированных инженерных расчётов (САПР, или CAE) и КЭ решения линейных и нелинейных, стационарных и нестационарных пространственных задач механики деформируемого твёрдого тела и механики конструкций (включая нестационарные геометрически и физически нелинейные задачи контактного взаимодействия элементов конструкций), задач механики жидкости и газа, теплопередачи и теплообмена, электродинамики, акустики, а также механики связанных полей.
* **Система электронного архива** — система структурированного хранения **электронных** документов, обеспечивающая надежность хранения, конфиденциальность и разграничение прав доступа, отслеживание истории использования документа, быстрый и удобный поиск. Распространённым в РФ является комплект ПО от Intermech, (Search, Techcard, AVS).

Как показала практика, около половины сотрудников, работающих с современным ПО, практически не используют весь его потенциал, не понимают требований стандартов и основных принципов создания электронной документации, совершают ошибки, которые после приходится исправлять, затрачивая на это немало времени. Грамотно сделанная и правильно оформленная документация, сданная в электронный архив, не требует постоянных изменений, на которые также требуется много времени. Главные причины: отсутствие продолжительных обучающих курсов, интеграция и синхронизация ПО между собой, интерфейсы на иностранных языках. Увеличение времени на разработку, проектирование, математические расчеты является серьезной проблемой при ОКК и замедляет весь процесс. Зачастую сложно найти специалистов, полностью разбирающихся в современном ПО, также существует ряд проблемы с организацией занятий с ними.

Для простоты построения обучающих материалов удобно будет добавлять в виде файлов с расширением .docx. Тестирующий модуль проще всего представить в виде тестов с принципом закрытых вопросов, работающий с .mdb. Вся информация добавляется в папку «study», в которой уже редактором создаются папки по темам. Запускаются программы исполняемым приложением .exe.

Исходя из этого, можно выделить основной функционал, поставленные перед разработкой системы:

* реализовать простой и интуитивно понятный интерфейс для пользователя без потери функционала программного обеспечения
* создать простой редактор, благодаря которому преподаватель сможет сформировать обучающий курс, загрузить его и, по необходимости, добавить тесты или проверочные задания для оценки полученных знаний.
* Основные направления обучения связаны с повышением квалификации в работе с CAD и CAE системами, электронным архивом и требованием стандартов. Нет необходимости усложнять тестирующий модуль, задания лучше всего выполнять самостоятельно.
* Подобные направления сложны, так что одним из преимуществ ЭОС будет возможность вставлять наглядные графические и видео примеры.

# **2 Тип и структура ОС**

В настоящее время, как наиболее оптимальный вариант в большинстве курсов программированного обучения, применяют текстовое и графическое представление информации. Классификация АОС по алгоритмическому построению представлена на рисунке 2.



Рис. 2. Классификация АОС по принципам алгоритмического построения

При использовании линейных алгоритмов АОС (рисунок 3) учащемуся, согласно методике, последовательно предъявляются слайды, заложенные в АОС. В качестве достоинств линейного алгоритма АОС можно отметить простоту разработки такой системы, а в качестве недостатков - трудоемкость раскрытия некоторых тем и невозможность гарантированного закрепления полученных знаний.

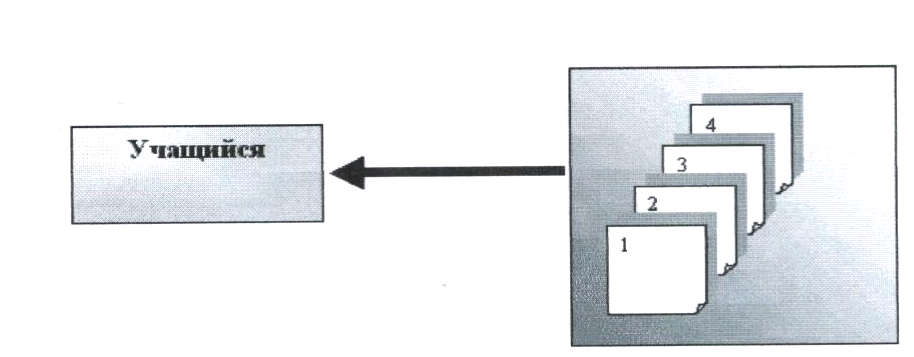


Рис. 3. Линейный алгоритм ОС

## **2.1 Выбор принципа построения ОС**

Основанием для классификации служат обычно особенности учебной деятельности обучаемых при работе с программами. Многие авторы выделяют четыре типа обучающих программ:

• тренировочные и контролирующие;

• наставнические;

• имитационные и моделирующие;

• развивающие игры.

Программы 1-го типа (тренировочные) предназначены для закрепления умений и навыков. Предполагается, что теоретический материал уже изучен. Эти программы в случайной последовательности предлагают учащемуся вопросы и задачи и подсчитывают количество правильно и неправильно решенных задач (в случае правильного ответа может выдаваться поощряющая ученика реплика). При неправильном ответе ученик может получить помощь в виде подсказки.

Программы 2-го типа (наставнические) предлагают ученикам теоретический материал для изучения. Задачи и вопросы служат в этих программах для организации человеко-машинного диалога, для управления ходом обучения. Так если ответы, даваемые учеником, неверны, программа может «откатиться назад» для повторного изучения теоретического материала.

Программы наставнического типа являются прямыми наследниками средств программированного обучения 60-х годов в том смысле, что основным теоретическим источником современного компьютерного или автоматизированного обучения следует считать программированное обучение. В публикациях зарубежных специалистов и сегодня под термином «программированное обучение» понимают современные компьютерные технологии. Одним из основоположников концепции программированного обучения является американский психолог Б.Ф.Скиннер.

Главным элементом программированного обучения является программа, понимаемая как упорядоченная последовательность рекомендаций (задач), которые передаются с помощью дидактической машины или программированного учебника и выполняются обучаемыми. Существует несколько известных разновидностей программированного обучения.

Ввиду чрезвычайно высокой трудоемкости написания программ такого рода на языках программирования и высоких требований к программистской квалификации разработчиков, они часто разрабатываются с использованием программных оболочек автоматизированных учебных курсов, имеющих свой язык программирования, интерфейс, рассчитанный на разработчика-непрограммиста.

Автоматизированные электронные обучающие системы представляют собой программно-технический комплекс, включающие в себя методическую, учебную и организационную поддержку процесса обучения, проводимого на базе информационных технологий.

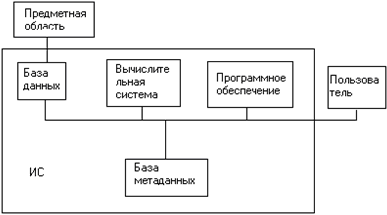


Рис.4. Общее представление электронных обучающих систем

Проектируя модель ЭОС необходимо сразу определить задачи, такие как:

* уяснение задачи, в результате чего определяется проблемная ситуация, устанавливается структура данных и неизвестных;
* разработка математической модели ситуации;
* составление плана решения (в виде текста, блок-схемы, графа и т.п.);
* выбор (разработка) алгоритма, наиболее эффективного для применения компьютера;
* составление программы в соответствии с алгоритмом;
* отладка и тестирование программы;
* проверка пригодности модели;
* оценка эффективности принятого плана решения;
* организация взаимодействия с компьютером.

Важно отметить, что лучшие обучающие системы не насыщенны сложными данными, формулами, схемами, совмещены с основным обучением, а сам процесс нередко дополняют забавные фразы, шутки, игры. Анализ и практика показывают, что положительные эффекты обучения наиболее отчетливо проявляются при:

* изучении базиса дисциплины, ее сложных закономерностей и алгоритмов, динамических процессов;
* реализации игр и имитаций;
* организации исследовательских и тренирующих процессов;
* автоматизации самоконтроля, контроля, оценки обучения;

Так как важно понимание основных принципом является требованием, а пользователи будущей системы обучения – специалисты, имеющие большой стаж работы и приверженцы консервативных методов проектирования, стоит учесть некоторые моменты:

* дидактический материал следует делить на незначительные дозы, шагами, которые обучаемые преодолевают относительно легко, шаг за шагом;
* вопросы, содержащиеся в отдельных рамках программы, не должны быть очень трудными, чтобы обучаемые не потеряли интереса к работе;
* обучаемые сами дают ответы на вопросы, привлекая для этого необходимую информацию;
* в ходе обучения учащихся сразу же информируют о том, правильны или ошибочны их ответы;
* все обучаемые проходят по очереди все рамки программы, но каждый делает это в удобном ему темпе;
* во избежание механического запоминания информации одна и та же мысль повторяется в различных вариантах и нескольких рамках программы.

Таким образом рациональнее выбрать принцип построение обучающей системы как наставнический, т.е. предлагают ученикам теоретический материал для изучения. Задачи и вопросы служат в этих программах для организации человеко-машинного диалога. Главная причина выбора подобного принципа обучения основано на том, чтобы заинтересовать обучающегося, помочь ему в трудных вопросах и не ограничивать в выборе материала.

## **2.2 Выбор структуры ОС**

Для непрерывного образования можно выделить задачи, которые решаются ЭОС на всех этапах формирования компетенций обучающихся. Такими задачами являются:

* регистрация пользователей;
* получение доступа к персональной среде;
* просмотр базы знаний с обеспечением целостности данных, исключением ошибок ввода, облегчением ввода данных, автоматизацией обработки описаний на множестве jбъектов и поиском;
* просмотр, наполнение и редактирование репозитория с широкими возможностями в оформлении учебного материала, большим набором мультимедийного наполнения, простотой и удобством, как создания новых учебных статей, так и их редактирования, с обеспечением коллективного доступа, наличием механизма ревизии описаний;
* ввод текущих оценок компетенции;
* оценка уровня знаний;
* контроль получения знаний;
* формирование индивидуальных траекторий – планирование индивидуальной программы обучения;
* реализация индивидуальных программ обучения с использованием индивидуальной среды обучения.

Для решения выделенных задач в интеллектуальной обучающей системе на основе агентно-ориентированного подхода можно выделить следующие компоненты:

* интерфейс обучаемого;
* модуль оценки знаний обучаемого;
* подсистема формирования индивидуальных планов обучения;
* cистема управления базой знаний;
* индивидуальная среда обучения;
* репозиторий учебных объектов.

Такая ИОС ориентирована на индивидуальную работу с обучаемыми.

Благодаря наличию практически на любом предприятии обширной локальной сети лучшее решение - использовать отдельный сервер, редактор новых обучающих материалов с возможностью загрузки на сервер и сам клиент ОС.

Это позволит администрировать ЭОС, легко добавлять/удалять/редактировать учебный материал, редактировать права, создавать учетные записи, а также контролировать усваивание материала, организовывать обратную связь. В любом случае потребуются специалисты, которым необходимо будет устраивать дополнительные занятия и консультации. В этом случае следует разделить ОС на два отдельных приложения: приложение для администрирования и контроля, редактор новых обучающих материалов с возможностью загрузки на сервер и сам клиент ОС.

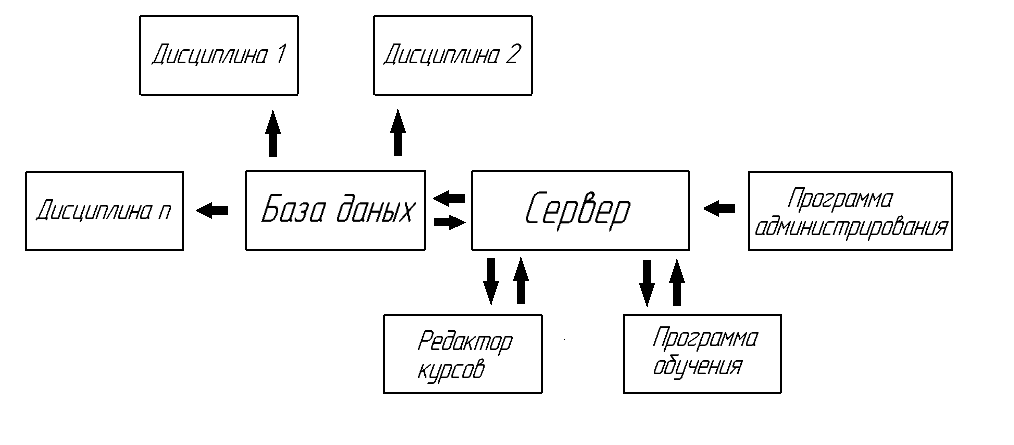


Рис.5. Упрощенная схема взаимодействия блоков

Администрирование программы включает в себя:

* Все материалы и курсы будут создаваться самостоятельно преподавателями.
* Сервер организован по принципу сервера базы данных.
* Администратор ЭОС единственный будет иметь доступ к статистике работы системы и результатам, а также управлять правами на доступ и изменение материалов.

Данное решение позволит упростить управление учетными записями, правами доступа к базе данных и редактированию обучающих материалов, изменение и совершенствование функционала ПО.

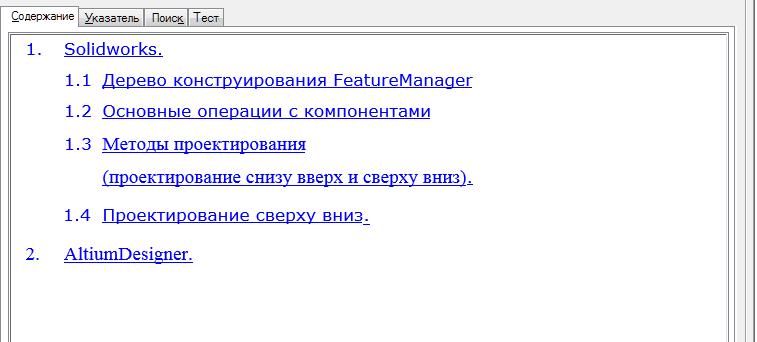


Рис.6. Упрощенный вид содержания

# **3 Обоснования выбранного языка программирования**

В качестве среды разработки выбрана Microsoft Visual Studio. линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight.

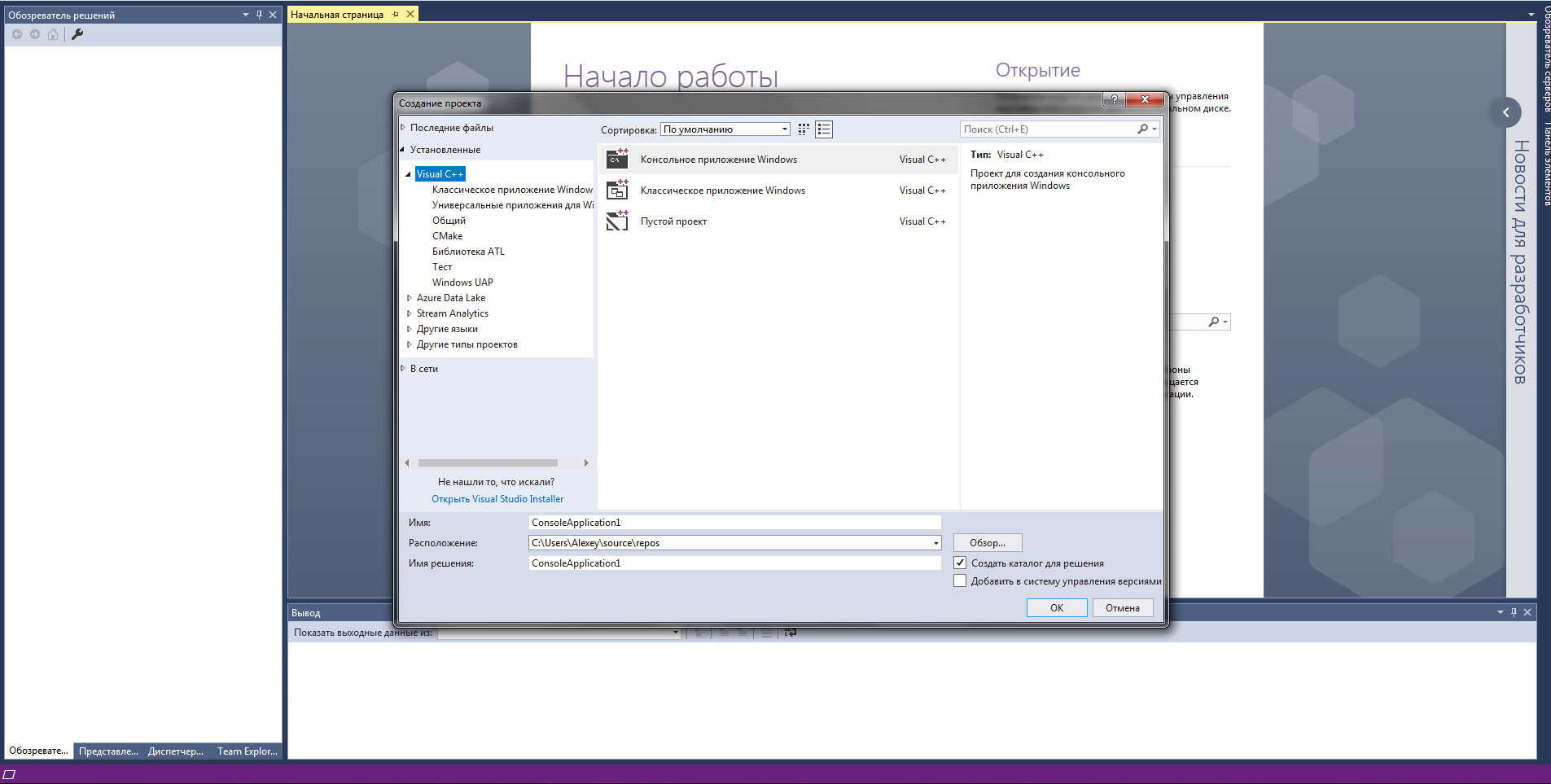


Рис.7. Microsoft Visual Studio 2017

Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как, например, Subversion и Visual SourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования) или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server).

Для реализации алгоритма в качестве основного был выбран язык программирования С++, который стал едва ли не стандартом для написания любых достаточно сложных программ. С++ - это универсальный язык программирования, задуманный так, чтобы сделать программирование более простым и приятным для серьёзного программиста. За исключением второстепенных деталей С++ является надмножеством языка программирования C (его объектно-ориентированным расширением). Основной целью разработчиков C было создание языка, который вместо неудобного и лишённого наглядности ассемблера применялся бы в самых насущных задачах системного программирования. К достоинствам языка С относятся лаконичность записи алгоритмов, логическая стройность написанных на нём программ, их переносимость между компьютерами с различной архитектурой и различными операционными средами (чего не скажешь о программах на машинно-зависимом языке ассемблера). Компиляторы языка разработаны практически для всех существующих в настоящее время платформ.

*C++* – универсальный язык общего назначения, область приложения которого – программирование систем в самом широком смысле. Кроме того, *C++* успешно использовался во многих приложениях, которые не укладываются в эти рамки. Реализация *C++* осуществлена для машин в диапазоне от самых современных микрокомпьютеров до самых мощных суперкомпьютеров и для почти всех операционных систем.

За исключением второстепенных деталей, *C++* является надмножеством языка программирования *С*. Помимо возможностей, которые даёт *С*, *C++* предоставляет гибкие и эффективные средства определения новых типов. Используя определения новых типов, точно отвечающих концепциям прикладной области, программист может разделять разрабатываемую программу на легко поддающиеся контролю части. Такой метод построения программ часто называют абстракцией данных. Информация о типах содержится в некоторых объектах типов, определённых пользователем. Такие объекты просты и надёжны в использовании в тех ситуациях, когда тип нельзя установить на стадии компиляции. Программирование с применением таких объектов часто называют объектно-ориентированным. При правильном использовании этот метод даёт более короткие, проще понимаемые и легче контролируемые программы.

Одним из первоначальных предназначений *С* было применение его вместо программирования на ассемблере в самых насущных задачах системного программирования. Когда проектировался *C*++, были приняты меры, чтобы не ставить под угрозу успехи в этой области. Особое внимание, уделенное структуризации при разработке *C*++, отразилось на возрастании масштаба программ, написанных со времени разработки *С*. Маленькую программу (меньше 1000 строк) вы можете заставить работать с помощью грубой силы, даже нарушая все правила хорошего стиля. Для программы больших размеров это не совсем так. Если программа в 10000 строк имеет плохую структуру, то вы обнаружите, что новые ошибки появляются так же быстро, как появляются старые. *С*++ был разработан так, чтобы дать возможность разумным образом структурировать большие программы так, чтобы для одного человека не было непомерной проблемой справляться с программами в 25000 строк.

Существуют программы гораздо больших размеров, однако те из них, которые работают, как оказывается, состоят из большого числа почти независимых частей, каждая из которых намного меньше указанных пределов. Естественно, сложность написания и поддержки программы зависит от сложности разработки, а не просто от числа строк текста программы, так что точные цифры, с помощью которых были выражены предыдущие соображения, не следует воспринимать слишком буквально.

Не каждая часть программы, однако, может быть хорошо структурирована, независима от аппаратного обеспечения, легко читаема и т.п. *С*++ обладает возможностями, предназначенными для того, чтобы непосредственно и эффективно работать с аппаратными средствами, не беспокоясь о безопасности и простоте понимания. Он также имеет возможности, позволяющие скрывать такие детали за элегантными и надёжными интерфейсами.

Естественно, что создание очень больших пакетов программ подразумевает использование *C*++ целыми группами программистов. Здесь в полную силу работают модульность, строго типизированный интерфейс и гибкость *C*++. В *C*++ имеется также гармоническое сочетание средств для написания больших программ, как и в любом другом языке. Однако по мере того, как программы становятся всё больше, трудности, связанные с их разработкой и сопровождением, преобразуются из чисто языковых в более глобальные методические и управленческие проблемы.

# **Заключение**

В результате проделанной работы были выполнены следующие задачи:

* Выбран необходимый функционал будущей программы.
* Определены наиболее подходящие тип и структура ОС.
* Выбрана среда разработки и язык программирования.

В качестве среды разработки был выбран Microsoft Visual Studio, в качестве языка программирования – C++ за его модульность, строго типизированный интерфейс и гибкость. Основная сфера применения приложения – институты и предприятия, основная цель – повышение квалификации специалистов при работе с CAD и CAE системами, обучение стандартам электронного документооборота. Благодаря наличию практически на любом предприятии обширной локальной сети решено использовать отдельный сервер и два приложения для работы с ним – редактор и сама ЭОС. Программа будет разделена на несколько модулей, каждый из которых будет выполнять свою задачу. С помощью главного модуля программы пользователь будет осуществлять навигацию по обучающей системе. Каждый этап главного модуля будет содержать информационные подсказки пользователю, полученные результаты, и тесты на общую проверку знаний. Так же главный модуль программы будет содержать алгоритм проверки на количество запущенных одновременно копий обучающей системы, что не позволит запускать больше одной копии на одном компьютере. После завершения работы программы, введенный пользователем вариант исходных данных будет сохранен и загружен при следующем запуске программы, что удобно для обучающихся. Редактор реализован отдельным приложением, имеет простые функции «добавить/удалить/редактировать». Учебный материал добавляется в виде .docx файлов. Тестирующий модуль представлен в виде тестов с принципом закрытых вопросов, работающий с .mdb. Вся информация добавляется в папку «study», в которой уже редактором создаются папки по темам. Запускаются программы исполняемым приложением .exe. Последний этап разработки – проверка функциональности и отладка ПО.

По мере выполнения работы были выработаны собственные рекомендации по разработке структуры и выбора функционала обучающей системы. Так как разработка обучающей системы только начинается, принято решение сформировать стандарт предприятия на построение подобных систем. Улучшение модуля тестирования, редактирования, а также представления информации поставлены как дальнейшие цели по повышению качества обучающей системы. Создание тестов по принципу открытых вопросов, то есть таких, на которые обучающийся должен сам формулировать ответ без каких-либо синтаксических ограничений также в планах улучшения электронной обучающей системы.

# **Список использованных источников**

1. Сайт «Техэксперт». Справочная система [Электронный ресурс]. Режим доступа: <http://www.cntd.ru/> Заглавие с экрана. – (Дата обращения 14.04.2018).
2. Сайт «OpenNet». Основы SQL.[Электронный ресурс]. Режим доступа: <https://www.opennet.ru/docs/RUS/rusql/>. – Заглавие с экрана. – (Дата обращения 14.04.18).
3. Сайт «Иторум». А.А. Воронцов. Классификация обучающих систем. [Электронный ресурс]. Режим доступа: http://wiki.itorum.ru/2011/04/klassifikaciya-obuchayushhix-sistem/. – Заглавие с экрана. – (Дата обращения 18.04.18).
4. Пак Н.И. Нелинейные технологии обучения в условиях информатизации: Учебное пособие. Красноярск, Изд-во РИО КГПУ. 1999.
5. Попов Э.В. Экспертные системы: решение неформализованных задач в диалоге с ЭВМ.- М.: Наука. 1987г.-288с
6. Трембач В.М. Системы управления базами эволюционирующих знаний для решения задач непрерывного образования: монография. – М.: МЭСИ, 2013
7. Рыбина Г.В., Паронджанов С.С. Моделирование процессов взаимодействия интеллектуальных агентов в многоагентных системах // Искусственный интеллект и принятие решений. – 2012. – № 1. – С. 3–15.
8. Агопонов С.В. Средства дистанционного обучения – СПб.: БХВ-Петербург, 2005. – 109 с.
9. Гусятников В.Н., Безруков А.И., Соколова Т.Н. Инструменты интеграции систем управления качеством в образовательный процесс // Интеграция образования. 2010. № 4. С. 16–19
10. Изотова В.Ф., Изотова Е.В. Информационное взаимодействие власти и общества: повышение роли официального сайта // Вестник Саратовской государственной юридической академии. 2013. № 1 (90). С. 210–215;