



# Курсовая работа по **.NET Framework**

**Версия 2.0.0**

## **Введение**

В рамках процесса обучения в Компьютерной Академии «Шаг» студенты направления «Разработка программного обеспечения», кроме экзаменов по предметам, также сдают курсовые по тому или иному предмету.

Давайте зададимся вопросом: почему в программе обучения есть курсовые? Есть несколько ответов на этот вопрос. Первый ответ: студентам важно поработать над комплексным проектом, объединяющим набор технологий. Например, если мы говорим о курсовом по .NET Framework, то в рамках проекта можно и нужно использовать максимальный спектр технологий .NET Framework. Второй ответ: студентам по окончании обучения необходимо иметь своё портфолио и курсовые работы отлично помогают в этом.

## **Выбор темы курсовой работы**

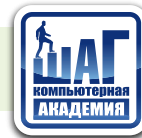
Компьютерная Академия «Шаг» и преподаватель, который ведет конкретный курс, не обязаны выдавать студенту тему курсовой работы. Мы можем предложить тему, но не обязаны делать это. Крайне важно стимулировать студента выбрать тему самому. Почему?

Если человек выбирает проект, который ему интересен по той или иной причине, мотивация работы над проектом будет гораздо выше. Если выбранный проект будет реальным коммерческим заказом, мы получаем идеальный курсовой проект. Безусловно, тема, предложенная в качестве курсовой, должна соответствовать тематике курсового.

## **Утверждение темы курсовой работы**

Тема курсовой работы студента должна быть утверждена заведующим кафедрой за один-два месяца до даты защиты. Студент описывает тему работы и список возможностей курсового проекта в заявке объемом не более двух листов.

На первом шаге полученную заявку утверждает преподаватель, отвечающий за реализацию курсового проекта. Например, если мы говорим о курсовом по .NET Framework, это тот преподаватель, который вел блок предметов .NET



Framework. На втором шаге тему утверждает заведующий кафедрой «Разработка программного обеспечения» филиала. На третьем шаге заявку студента получает менеджер учебного процесса конкретного направления. Заявка оформляется в двух экземплярах. Один экземпляр остается у студента, второй – у менеджера учебного процесса. Данный механизм может быть реализован и в электронном виде. Например, менеджер учебного процесса хранит отсканированные заявки с подписями.

Основная задача в процессе утверждения – убедиться в том, что предлагаемый проект по сложности, технологиям и полноте соответствует курсовой работе. Наличие подписей преподавателя и заведующего кафедрой показывает, что они проанализировали тему и уверены в том, что курсовой проект подходит по озвученным критериям. Если в филиале нет заведующего кафедрой, тогда подпись ставит только преподаватель.

Обращаем ваше внимание на то, что курсовая работа может быть выполнена одним студентом или командой студентов. В командном проекте, объём работы каждого студента должен быть равным объёму курсовой для одного студента.

## **Работа над курсовой работой**

Студент выполняет курсовую работу дома, вне стен Академии. Если у студента возникают вопросы, то в процессе работы он задаёт их преподавателю в рамках обучения.

В учебном плане стационара и ЕКО выделено время на консультации и защиту курсового проекта. Выделенные пары желательно расположить не на одной неделе, а раскидать на разные недели, для того чтобы у студента процесс подготовки был плавным. Вопросом распределения консультационных пар и датой защиты курсового занимается менеджер учебного процесса.

На полустационаре выделенное время в учебном плане предназначено только для защиты проекта.

## **Защита курсового проекта**

Дату защиты курсового проекта назначает менеджер учебного процесса, после консультаций с преподавателями и заведующим кафедрой.

На защите должны присутствовать менеджер учебного процесса направления (по возможности), преподаватель, отвечающий за курсовой студента, заведующий кафедрой. Заведующий кафедрой приходит на защиту с заявками студентов, которые хранились у менеджера учебного процесса.



Студент на защиту готовит распечатанный документ, в состав которого входит:

1. Титульный лист с названием темы, названием курсового проекта, ФИО студента;
2. Копия заявки, которая осталась у студента;
3. UML-диаграммы проекта. Обязательным условием является наличие диаграммы классов, диаграмма вариантов использования (use case diagram). Другие диаграммы прикладываются по желанию.
4. Схема базы данных (если в проекте есть база данных);
5. Исходный код курсового проекта (если студент не готов отдать исходный код проекта по какой-либо причине, то тогда ключевые куски кода).

Защита проводится в формате презентации. Студент рассказывает о своей работе перед комиссией. Использование презентаций в формате PowerPoint и других средств приветствуется. В рамках презентации студент должен рассказать о своем проекте, почему была выбрана данная тема, осветить наличие конкурентов, обосновать выбранные технологии и т.д. После рассказа о своей работе студент должен продемонстрировать работу проекта на практике.

Комиссия может задавать абсолютно любые вопросы по проекту, в рамках процесса защиты. Если комиссия захочет задать вопрос по исходному коду, студент обязан продемонстрировать место в коде, которое интересует комиссию.

Максимальная длительность выступления студента 15 минут, рекомендуемое время для вопросов комиссии 15 минут. Итого: общее время работы с конкретным студентом на защите 30 минут.

После выступления всех студентов комиссия устраивает коллегиальное обсуждение оценок (без присутствия студентов). Оценка конкретному студенту выставляется при достижении коллегиального согласия или большинства. После выставления всех оценок, заведующий кафедрой объявляет их перед студентами сразу после финального совещания. Если у студента возникает вопрос по выставленной оценке, заведующий кафедрой обязан ответить индивидуально студенту после объявления результатов.

Студенты, не сдавшие курсовой или желающие получить более высокую оценку, отправляются на пересдачу. Дату пересдачи назначает менеджер учебного процесса после консультаций с преподавателями и заведующим кафедрой.



## Примеры тем для курсового проекта по .NET Framework

### 1. Чат на основе протокола UDP

- Краткое описание: чат на основе протокола UDP. В чате должна быть история сообщений, список контактов, приватная, групповая и массовая рассылка, черный список, изменение состояний пользователей.
- Рабочая среда: локальная сеть.
- Пользователи: клиенты локальной сети.

### 2. Чат на основе протокола TCP

- Краткое описание: чат на основе протокола TCP. В чате должна быть история сообщений, список контактов, приватная, групповая и массовая рассылка, черный список, изменение состояний пользователей. У чата должен быть сервер, написанный на WCF, который управляет работой чата, позволяет администратору менять настройки.
- Рабочая среда: локальная сеть.
- Пользователи: клиенты локальной сети.

### 3. Тестовая программа

- Краткое описание: программа для проведения тестирования. Тестовые задания и список пользователей хранится в базе. Также ведется учет каждого прохождения. Программа должна поддерживать различные виды вопросов и загрузку картинок к вопросам. Программа делится на две части: редактор тестов, тестовая система. В редакторе тестов создаются тесты, которые хранятся в базе. Каждый тест состоит из набора вопросов. Вопрос содержит произвольное количество вариантов ответов. Вопросы бывают двух видов: выбор одного варианта из многих, выбор нескольких вариантов. Каждый вопрос имеет свой вес. К вопросу можно загружать картинку. Тестовая система воспроизводит тесты для пользователей. Пользователь заходит в систему под своим логином и паролем. Он имеет ограниченное количество попыток для тестирования. Результаты заносятся в базу. Разработанное приложение подключается к удаленной базе (SQL Server 2008) и функционирует на основе информации записанной в ней (тесты, пользователи, прохождения). Клиентские приложения тестовой системы не подключаются к базе напрямую. Они получают вопросы и отправляют ответы на сервер, который работает с базой.
- Рабочая среда: локальная сеть.
- Пользователи: ученики, преподаватели.

### 4. Программа для слежения за активностью пользователей сети.

### 5. Клавиатурный шпион с отправкой данных по сети на сервер.

### 6. Программа для ограничения запуска некоторых программ с изменением настроек по сети.