

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 15.06.2023 10:11:51
Уникальный программный ключ:
0b817ca911e6668abb13a5d426d39e5f1c11eab071e0430e448d1a56109

МИНОБРАЗОВАНИЯ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра информационных систем и технологий

УТВЕРЖДАЮ
Проректор по учебной работе
О. Г. Локтионова
«25» 01 2018г.



ОПИСАНИЕ И АНАЛИЗ ПРОЕКТА ИС

Методические указания по выполнению
практических работ по дисциплине
«Проектирование информационных систем»
для студентов направления подготовки бакалавров
09.03.02 Информационные системы
09.03.03 Прикладная информатика

Курск 2018

УДК 004.82 (075.8)

Составитель Т.И.Лапина

Рецензент

Кандидат технических наук, доцент Е.А.Петрик

Описание и анализ проекта ИС: методические указания по выполнению практических работ по дисциплине «Проектный практикум» / Юго-Зап. гос. ун-т; сост.: Т. И. Лапина, Курск, 2018. 69с. ил. 44, табл.5, Билиогр.: с.69.

Содержат краткие теоретические сведения о методах разработки требований к проекту информационных систем.

Методические указания соответствуют требованиям программ по направлениям подготовки бакалавров: 09.03.02 Информационные системы, 09.03.03 Прикладная информатика

Предназначены для студентов направления подготовки бакалавров 09.03.02 Информационные системы, 09.03.03 Прикладная информатика дневной и заочной форм обучения.

Текст печатается в авторской редакции

Подписано в печать 25.01.18 Формат 60x84 1/16.

Усл. печ. л. 1,7. Уч. - изд. л. 1,6. Тираж 100 экз. Заказ 190. Бесплатно.

Юго - Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

Практическая работа №1

Описание проекта информационной системы и анализ типовых проектных решений

1. Цель работы

Получить навыки описания и анализа требований к проектируемой информационной системе, методы организации коллектива и распределения ролей в группе разработчиков.

2. Основные теоретические положения

Практическая работа направлена на ознакомление с процессом описания информационной системы и получение навыков по использованию основных методов анализа ИС.

Требования к результатам выполнения практикума:

1. наличие описания информационной системы;
2. проведение анализа осуществимости выполнения проекта;
3. наличие заключения о возможности реализации проекта, содержащего рекомендации относительно разработки системы, базовые предложения по объёму требуемого бюджета, числу разработчиков, времени и требуемому программному обеспечению.

2.1 Общие сведения о разработке ИС

Проблемы управления программными проектами впервые проявились в 60-х - начале 70-х годов, когда провалились многие большие проекты по разработке программных продуктов. Были зафиксированы задержки в создании ПО, оно было ненадежным, затраты на разработку в несколько раз превосходили первоначальные оценки, созданные программные системы часто имели низкие показатели производительности. Причины провалов коренились в тех подходах, которые использовались в управлении проектами. Применяемая методика была основана на опыте управления техническими проектами и оказалась неэффективной при разработке программного обеспечения.

Важно понимать разницу между профессиональной разработкой ПО и любительским программированием. Необходимость управления программными проектами вытекает из того факта, что процесс создания профессионального ПО всегда является субъектом бюджетной политики организации, где оно разрабатывается, и имеет временные ограничения. Работа *руководителя программного проекта* по большому

счету заключается в том, чтобы гарантировать выполнение этих бюджетных и временных ограничений с учетом бизнес-целей организации относительно разрабатываемого ПО.

Руководители проектов призваны спланировать все этапы разработки программного продукта. Они также должны контролировать ход выполнения работ и соблюдения всех требуемых стандартов. Постоянный контроль за ходом выполнения работ необходим для того, чтобы процесс разработки не выходил за временные и бюджетные ограничения. Хорошее управление не гарантирует успешного завершения проекта, но плохое управление обязательно приведет к его провалу. Это может выразиться в задержке сроков сдачи готового ПО, в превышении сметной стоимости проекта и в несоответствии готового ПО спецификации требований.

Процесс разработки ПО существенно отличается от процессов реализации технических проектов, что порождает определенные сложности в управлении программными проектами:

1. *Программный продукт нематериален.* Программное обеспечение нематериально, его нельзя увидеть или потрогать. Руководитель программного проекта не видит процесс "роста" разрабатываемого ПО. Он может полагаться только на документацию, которая фиксирует процесс разработки программного продукта.

2. *Не существует стандартных процессов разработки ПО.* На сегодняшний день не существует четкой зависимости между процессом создания ПО и типом создаваемого программного продукта. Другие технические дисциплины имеют длительную историю, процессы разработки технических изделий многократно опробованы и проверены. Процессы создания большинства технических систем хорошо изучены. Изучением же процессов создания ПО специалисты занимаются только последнее время. Поэтому пока нельзя точно предсказать, на каком этапе процесса разработки ПО могут возникнуть проблемы, угрожающие всему программному проекту.

3. *Большие программные проекты - это часто "одноразовые" проекты.* Большие программные проекты, как правило, значительно отличаются от проектов, реализованных ранее. Поэтому, чтобы уменьшить неопределенность в планировании проекта, руководители проектов должны обладать очень большим практическим опытом. Но постоянные технологические изменения в компьютерной технике и коммуникационном оборудовании обесценивают предыдущий опыт. Знания

и навыки, накопленные опытом, могут не востребоваться в новом проекте.

Перечисленные отличия могут привести к тому, что реализация проекта выйдет из временного графика или превысит бюджетные ассигнования. Программные системы зачастую оказываются новинками как в "идеологическом", так и в техническом плане. Поэтому, предвидя возможные проблемы в реализации программного проекта, следует всегда помнить, что многим из них свойственно выходить за рамки временных и бюджетных ограничений.

2.2 Процесс управления разработкой программного обеспечения

Невозможно описать и стандартизировать все работы, выполняемые в проекте по созданию ПО. Эти работы весьма существенно зависят от организации, где выполняется разработка ПО, и от типа создаваемого программного продукта. Но всегда можно выделить следующие:

- Написание предложений по созданию ИС.
- Планирование и составление графика работ по созданию ИС.
- Оценивание стоимости проекта ИС.
- Контроль за ходом выполнения работ.

Первая стадия программного проекта может состоять из написания предложений по реализации этого проекта. Предложения должны содержать описание целей проектов и способов их достижения. Они также обычно включают в себя оценки финансовых и временных затрат на выполнение проекта. При необходимости здесь могут приводиться обоснования для передачи проекта на выполнение сторонней организации или команде разработчиков.

Написание предложений — очень ответственная работа, так как для многих организаций вопрос о том, будет ли проект выполняться самой организацией или разрабатываться по контракту сторонней компанией, является критическим. Не существует каких-либо рекомендаций по написанию предложений, многое здесь зависит от опыта.

На этапе *планирования проекта* определяются процессы, этапы и полученные на каждом из них результаты, которые должны привести к выполнению проекта. Реализация этого плана приведет к достижению целей проекта. Определение стоимости проекта напрямую связано с его планированием, поскольку здесь оцениваются ресурсы, требующиеся для выполнения плана.

Контроль за ходом выполнения работ (мониторинг проекта) — это непрерывный процесс, продолжающийся в течение всего срока реализации проекта. Руководитель должен постоянно отслеживать ход реализации проекта и сравнивать фактические и плановые показатели выполнения работ с их стоимостью. Хотя многие организации имеют механизмы формального мониторинга работ, опытный руководитель может составить ясную картину о стадии развитии проекта просто путем неформального общения с разработчиками.

Неформальный мониторинг часто помогает обнаружить потенциальные проблемы, которые в явном виде могут обнаружиться позднее. Например, ежедневное обсуждение хода выполнения работ может выявить отдельные недоработки в создаваемом программном продукте. Вместо ожидания отчетов, в которых будет отражен факт "пробуксовки" графика работ, можно обсудить со специалистами намечающиеся программистские проблемы и не допустить срыва графика работ.

В течение реализации проекта обычно происходит несколько формальных контрольных проверок хода выполнения работ по созданию ПО. Такие проверки должны дать общую картину хода реализации проекта в целом и показать, насколько уже разработанная часть ПО соответствует целям проекта.

Время выполнения больших программных проектов может занимать несколько лет. В течение этого времени цели и намерения организации, заказавшей программный проект, могут существенно измениться. Может оказаться, что разрабатываемый программный продукт стал уже ненужным либо исходные требования к создаваемому ПО просто устарели и их необходимо кардинально менять. В такой ситуации руководство организации-разработчика может принять решение о прекращении разработки ПО или об изменении проекта в целом с тем, чтобы учесть изменившиеся цели и намерения организации-заказчика.

Руководители проектов обычно обязаны сами *подбирать исполнителей* для своих проектов. В идеальном случае профессиональный уровень исполнителей должен соответствовать той работе, которую они будут выполнять в ходе реализации проекта.

Однако во многих случаях руководители должны полагаться на команду разработчиков, которая далека от идеальной. Такая ситуация может быть вызвана следующими причинами:

1. Бюджет проекта не позволяет привлечь высококвалифицированный персонал. В таком случае за меньшую плату привлекаются менее квалифицированные специалисты.

2. Бывают ситуации, когда невозможно найти специалистов необходимой квалификации как в самой организации-разработчике, так и вне ее. Например, в организации "лучшие люди" могут быть уже заняты в других проектах.

3. Организация хочет повысить профессиональный уровень своих работников. В этом случае она может привлечь к участию в проекте неопытных или недостаточно квалифицированных работников, чтобы они приобрели необходимый опыт и по-учились у более опытных специалистов.

Таким образом, почти всегда подбор специалистов для выполнения проекта имеет определенные ограничения и не является свободным. Вместе с тем необходимо, чтобы хотя бы несколько членов группы разработчиков имели квалификацию и опыт, достаточные для работы над данным проектом. В противном случае невозможно избежать ошибок в разработке ПО.

Руководитель проекта обычно обязан посылать *отчеты* о ходе его выполнения как заказчику, так и подрядным организациям. Это должны быть краткие документы, основанные на информации, извлекаемой из подробных отчетов о проекте. В этих отчетах должна быть та информация, которая позволяет четко оценить степень готовности создаваемого программного продукта.

В рамках курса «Технология разработки программного обеспечения» выделены следующие роли в группе по разработке ПО:

- Руководитель - общее руководство проектом, написание документации, общение с заказчиком ПО
- Системный аналитик - разработка требований (составление технического задания, проекта программного обеспечения)
- Тестер - составление плана тестирования и аттестации готового ПО (продукта), составление сценария тестирования, базовый пример, проведение мероприятий по плану тестирования
- Разработчик - моделирование компонент программного обеспечения, кодирование.

Планирование проекта разработки программного обеспечения

Эффективное управление программным проектом напрямую зависит от правильного планирования работ, необходимых для его выполнения. План помогает руководителю предвидеть проблемы, которые могут возникнуть на каких-либо этапах создания ПО, и разработать превентивные меры для их предупреждения или решения. План, разработанный на начальном этапе проекта, рассматривается всеми его

участниками как руководящий документ, выполнение которого должно привести к успешному завершению проекта. Этот первоначальный план должен максимально подробно описывать все этапы реализации проекта.

Процесс планирования начинается, исходя из описания системы, с определения проектных ограничений (временные ограничения, возможности наличного персонала, бюджетные ограничения и т.д.). Эти ограничения должны определяться параллельно с оцениванием проектных параметров, таких как структура и размер проекта, а также распределением функций среди исполнителей. Затем определяются этапы разработки и то, какие результаты документация, прототипы, подсистемы или версии программного продукта) должны быть получены по окончании этих этапов. Далее начинается циклическая часть планирования. Сначала разрабатывается график работ по выполнению проекта или дается разрешение на продолжение использования ранее созданного графика. После этого проводится контроль выполнения работ и отмечаются расхождения между реальным и плановым ходом работ.

Далее, по мере поступления новой информации о ходе выполнения проекта, возможен пересмотр первоначальных оценок параметров проекта. Это, в свою очередь, может привести к изменению графика работ. Если в результате этих изменений нарушаются сроки завершения проекта, должны быть пересмотрены (и согласованы с заказчиком ПО) проектные ограничения.

Конечно, большинство руководителей проектов не думают, что реализация их проектов пройдет гладко, без всяких проблем. Желательно описать возможные проблемы еще до того, как они проявят себя в ходе выполнения проекта. Поэтому лучше составлять "пессимистические" графики работ, чем "оптимистические". Но, конечно, невозможно построить план, учитывающий все, в том числе случайные, проблемы и задержки выполнения проекта, поэтому и возникает необходимость периодического пересмотра проектных ограничений и этапов создания программного продукта.

План проекта должен четко показать ресурсы, необходимые для реализации проекта, разделение работ на этапы и временной график выполнения этих этапов. В некоторых организациях план проекта составляется как единый документ, содержащий все виды планов, описанных выше. В других случаях план проекта описывает только техно-

логический процесс создания ПО. В таком плане обязательно присутствуют ссылки на планы других видов, но они разрабатываются отдельно от плана проекта.

Детализация планов проектов очень различается в зависимости от типа разрабатываемого программного продукта и организации-разработчика. Но в любом случае большинство планов содержат следующие разделы.

1. *Введение.* Краткое описание целей проекта и проектных ограничений (бюджетных, временных и т.д.), которые важны для управления проектом.

2. *Организация выполнения проекта.* Описание способа подбора команды разработчиков и распределение обязанностей между членами команды.

3. *Анализ рисков.* Описание возможных проектных рисков, вероятности их проявления и стратегий, направленных на их уменьшение.

4. *Аппаратные и программные ресурсы, необходимые для реализации проекта.* Перечень аппаратных средств и программного обеспечения, необходимого для разработки программного продукта. Если аппаратные средства требуется закупать, приводится их стоимость совместно с графиком закупки и поставки.

5. *Разбиение работ на этапы.* Процесс реализации проекта разбивается на отдельные процессы, определяются этапы выполнения проекта, приводится описание результатов ("выходов") каждого этапа и контрольные отметки.

1. *График работ.* В этом графике отображаются зависимости между отдельными процессами (этапами) разработки ПО, оценки времени их выполнения и распределение членов команды разработчиков по отдельным этапам.

2. *Механизмы мониторинга и контроля за ходом выполнения проекта.*

Описываются предоставляемые руководителем отчеты о ходе выполнения работ, сроки их предоставления, а также механизмы мониторинга всего проекта.

План должен регулярно пересматриваться в процессе реализации проекта. Одни части плана, например график работ, изменяются часто, другие более стабильны. Для внесения изменений в план требуется специальная организация документопотока, позволяющая отслеживать эти изменения.

2.3. Общие сведения о требованиях к информационным системам

Проблемы, которые приходится решать специалистам в процессе создания программного обеспечения, очень сложны. Природа этих проблем не всегда ясна, особенно если разрабатываемая программная система инновационная. В частности, трудно чётко описать те действия, которые должна выполнять система. Описание функциональных возможностей и ограничений, накладываемых на систему, называется требованиями к этой системе, а сам процесс формирования, анализа, документирования и проверки этих функциональных возможностей и ограничений - разработкой требований.

Требования подразделяются на пользовательские и системные. Пользовательские требования - это описание на естественном языке (плюс поясняющие диаграммы) функций, выполняемых системой, и ограничений, накладываемых на неё. Системные требования - это описание особенностей системы (архитектура системы, требования к параметрам оборудования и т.д.), необходимых для эффективной реализации требований пользователя.

2.4. Первые шаги по разработке требований к информационным системам - анализ осуществимости

Разработка требований - это процесс, включающий мероприятия, необходимые для создания и утверждения документа, содержащего спецификацию системных требований. Для новых программных систем процесс разработки требований должен начинаться с анализа осуществимости. Началом такого анализа является общее описание системы и ее назначения, а результатом анализа — отчет, в котором должна быть четкая рекомендация, продолжать или нет процесс разработки требований проектируемой системы. Другими словами, анализ осуществимости должен осветить следующие вопросы.

Отвечает ли система общим и бизнес-целям организации-заказчика и организации-разработчика?

Можно ли реализовать систему, используя существующие на данный момент технологии и не выходя за пределы заданной стоимости?

Можно ли объединить систему с другими системами, которые уже эксплуатируются?

Критическим является вопрос, будет ли система соответствовать целям организации. Если система не соответствует этим целям, она не представляет никакой ценности для организации. В то же время многие организации разрабатывают системы, не соответствующие их целям, либо не совсем ясно понимая эти цели, либо под влиянием политических или общественных факторов.

Выполнение анализа осуществимости включает сбор и анализ информации о будущей системе и написание соответствующего отчета. Сначала следует определить, какая именно информация необходима, чтобы ответить на поставленные выше вопросы. Например, эту информацию можно получить, ответив на следующее:

1. Что произойдет с организацией, если система не будет введена в эксплуатацию?
2. Какие текущие проблемы существуют в организации и как новая система поможет их решить?
3. Каким образом система будет способствовать целям бизнеса?
4. Требуется ли разработка системы технологии, которая до этого не использовалась в организации?

Далее необходимо определить источники информации. Это могут быть менеджеры отделов, где система будет использоваться, разработчики программного обеспечения, знакомые с типом будущей системы, технологи, конечные пользователи и т.д.

После обработки собранной информации готовится отчет по анализу осуществимости создания системы. В нем должны быть даны рекомендации относительно продолжения разработки системы. Могут быть предложены изменения бюджета и графика работ по созданию системы или предъявлены более высокие требования к системе.

Порядок выполнения работы

1. Изучить предлагаемый теоретический материал.
2. Составить подробное описание информационной системы.
3. На основании описания системы провести анализ осуществимости.

В ходе анализа ответить на вопросы

1. Что произойдет с организацией, если система не будет введена в эксплуатацию ?

2. Какие текущие проблемы существуют в организации и как новая система поможет их решить?

Каким образом система будет способствовать целям бизнеса? Требуется ли разработка системы технологии, которая до этого не использовалась в организации?

Результатом анализа должно явиться заключение о возможности реализации проекта.

4. Распределить роли в группе (руководитель проекта-разработчик, системный аналитик-разработчик, тестер-разработчик).

Заполнить разделы плана:

1. *Введение*

2. *Организация выполнения проекта*

3. *Анализ рисков*

Разделы должны содержать рекомендации относительно разработки системы, базовые предложения по объёму требуемого бюджета, числу разработчиков, времени и требуемому программному обеспечению.

3. Содержание отчета

В отчете следует указать:

1. Цель работы

2. Введение. Краткое описание целей проекта и проектных ограничений (бюджетных, временных и т.д.), которые важны для управления проектом

3. Описание информационной системы (ПО)

наличие заключения о возможности реализации проекта, содержащего рекомендации относительно разработки системы, базовые предложения по объёму требуемого бюджета, числу разработчиков, времени и требуемому программному обеспечению

4. Анализ осуществимости (согласно требованиям к результатам выполнения лабораторного практикума п.2), указать возможные проблемы и пути их решения.

5. Роли участников группы разработки ПО.

6. Программно-аппаратные средства, используемые при выполнении работы.

7. Заключение (выводы)

8. Список используемой литературы

4. Контрольные вопросы

1. Какие вопросы включает методика описания проектируемой ИС?
2. На каких уровнях проводится обследование аспектов деятельности предприятий?
3. Какие существуют универсальные методы, пригодные для обследования всех функциональных звеньев предприятия?
4. Какие существуют документы для описания ИС?
5. Что такое «реинжиниринг» объекта автоматизации?
6. С какой целью проводится «реинжиниринг» объекта автоматизации?
7. В каких направлениях выполняется информационный анализ предметной области?
8. Как систематизируется полученная информация?
9. Кто выполняет функции анализа объекта автоматизации?
10. Цель анализа полученной информации.

Список литературы

1. Лапина, Т. И. Информационные системы. Проектный практикум к выполнению и защите ВКР бакалавра по направлению 09.03.02 Информационные системы, 09.03.03 Прикладная информатика/ Т. И. Лапина//Юго-Западный гос. ун-т, ЗАО «Университетская книга»—Курск, 2016.—99с.
2. Золотов, С. Ю. Проектирование информационных систем [Электронный ресурс] : учебное пособие / С. Ю. Золотов. - Томск : Эль Контент, 2013. - 88 с.
3. Абрамов, Г. В. Проектирование информационных систем [Электронный ресурс] : учебное пособие / Г. В. Абрамов, И. Медведкова, Л. Коробова. - Воронеж : Воронежский государственный университет инженерных технологий, 2012. - 172 с.
4. Стасышин, В. М. Проектирование информационных систем и баз данных [Электронный ресурс] : учебное пособие / В. М. Стасышин. - Новосибирск : НГТУ, 2012. - 100 с.
5. .

Практическая работа №2

Разработка технического задания на проектирование информационной системы

1. Цель работы

Получение навыков анализа объекта автоматизации, выявления целей задач проектируемой ИС, разработки технического задания на проектирование информационной системы

2 Основные теоретические положения

Стандарт ГОСТ 34.602-89 распространяется на автоматизированные системы (ИС) для автоматизации различных видов деятельности (управление, проектирование, исследование и т. п.), включая их сочетания, и устанавливает состав, содержание, правила оформления документа “Техническое задание на создание (развитие или модернизацию) системы” (далее - ТЗ на ИС).

ТЗ на ИС является основным документом, определяющим требования и порядок создания (развития или модернизации - далее создания) автоматизированной системы, в соответствии с которым проводится разработка ИС и ее приемка при вводе в действие.

ТЗ на ИС разрабатывают на систему в целом, предназначенную для работы самостоятельно или в составе другой системы. Дополнительно могут быть разработаны ТЗ на части ИС:

- на подсистемы ИС, комплексы задач ИС и т. п. в соответствии с требованиями настоящего стандарта;
- на комплектующие средства технического обеспечения и программно-технические комплексы в соответствии со стандартами ЕСКД и СРПП;
- на программные средства в соответствии со стандартами ЕСПД;
- на информационные изделия в соответствии с ГОСТ 19.201 и НТД, действующей в ведомстве заказчика ИС.

Требования к ИС в объеме, установленном настоящим стандартом, могут быть включены в задание на проектирование вновь создаваемого объекта автоматизации. В этом случае ТЗ на ИС не разрабатывают.

Включаемые в ТЗ на ИС требования должны соответствовать современному уровню развития науки и техники и не уступать аналогичным требованиям, предъявляемым к лучшим современным отечественным и зарубежным аналогам. Задаваемые в ТЗ на ИС требования не должны ограничивать разработчика системы в поиске и реализации наиболее эффективных технических, технико-экономических и других решений.

ТЗ на ИС разрабатывают на основании исходных данных в том числе содержащихся в итоговой документации стадии “Исследование и обоснование создания ИС”, установленной ГОСТ 24.601.

В ТЗ на ИС включают только те требования, которые дополняют требования к системам данного вида (АСУ, САПР, АСНИ и т. д.), содержащиеся в действующих НТД, и определяются спецификой конкретного объекта, для которого создается система.

ТЗ на ИС содержит следующие разделы, которые могут быть разделены на подразделы:

№	Реальные условия	Курсовой проект
1.	общие сведения	
2.	назначение и цели создания (развития) системы	
3.	характеристика объектов автоматизации	
4.	требования к системе	
5.	состав и содержание работ по созданию системы	
6.	порядок контроля и приемки системы	
7.	требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие	
8.	требования к документированию	
9.	источники разработки	
10.	приложения	

В зависимости от вида, назначения, специфических особенностей объекта автоматизации и условий функционирования системы допускается оформлять разделы ТЗ в виде приложений, вводить дополнительные, исключать или объединять подразделы ТЗ.

Рассмотрим иерархию до 5 уровней.

СОСТАВ – Разделы (10) – Подразделы – Группы требований – требования/условия

В разделе **“Общие сведения”** указывают:

- 1) полное наименование системы и ее условное обозначение;
- 2) шифр темы или шифр (номер) договора;
- 3) наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты;
- 4) перечень документов, на основании которых создается система, кем и когда утверждены эти документы;
- 5) плановые сроки начала и окончания работы по созданию системы;
- 6) сведения об источниках и порядке финансирования работ;
- 7) порядок оформления и предъявления заказчику результатов работ по созданию системы (ее частей), по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы.

Раздел **“Назначение и цели создания (развития) системы”** состоит из подразделов:

В подразделе **“Назначение системы”** указывают вид автоматизируемой деятельности (управление, проектирование и т. п.) и перечень объектов автоматизации (объектов), на которых предполагается ее использовать. Для АСУ дополнительно указывают перечень автоматизируемых органов (пунктов) управления и управляемых объектов. В подразделе **“Цели создания системы”** приводят наименования и требуемые значения технических, технологических, производственно-экономических или других показателей объекта автоматизации, которые должны быть достигнуты в результате создания ИС, и указывают критерии оценки достижения целей создания системы.

В разделе **“Характеристики объекта автоматизации”** приводят краткие сведения об объекте автоматизации или ссылки на документы, содержащие такую информацию и сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды.

Примечание: Для САПР в разделе дополнительно приводят основные параметры и характеристики объектов проектирования.

Раздел **“Требования к системе”** состоит из следующих подразделов:

- 1) требования к системе в целом;
- 2) требования к функциям (задачам), выполняемым системой;
- 3) требования к видам обеспечения.

Состав требований к системе, включаемых в данный раздел ТЗ на ИС, устанавливаются в зависимости от вида, назначения, специфических особенностей и условий функционирования конкретной системы. В каждом подразделе приводят ссылки на действующие НТД, определяющие требования к системам соответствующего вида.

В подразделе **“Требования к системе в целом”** указывают: требования к структуре и функционированию системы; требования к численности и квалификации персонала системы и режиму его работы;

показатели назначения;

требования к надежности;

требования безопасности;

требования к эргономике и технической эстетике;

требования к транспортабельности для подвижных ИС;

требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы;

требования к защите информации от несанкционированного доступа;

требования по сохранности информации при авариях;

требования к защите от влияния внешних воздействий;

требования к патентной чистоте;

требования по стандартизации и унификации;

дополнительные требования.

В требованиях к структуре и функционированию системы приводят:

1) перечень подсистем, их назначение и основные характеристики, требования к числу уровней иерархии и степени централизации системы;

2) требования к способам и средствам связи для информационного обмена между компонентами системы;

3) требования к характеристикам взаимосвязей создаваемой системы со смежными системами, требования к ее совместимости, в том числе указания о способах обмена информацией (автоматически, пересылкой документов, по телефону и т. п.);

4) требования к режимам функционирования системы;

5) требования по диагностированию системы;

6) перспективы развития, модернизации системы.

В требованиях к численности и квалификации персонала да ИС приводят:

требования к численности персонала (пользователей) ИС;

требования к квалификации персонала, порядку его подготовки и контролю знаний и навыков;

требуемый режим работы персонала ИС.

В требованиях к показателям назначения ИС приводят значения параметров, характеризующие степень соответствия системы ее назначению.

Для АСУ указывают:

степень приспособляемости системы к изменению процессов, методов управления, к отклонениям параметров объекта управления;

допустимые пределы модернизации и развития системы;

вероятностно-временные характеристики, при которых сохраняется целевое назначение системы.

В требованиях к надежности включают:

1) состав и количественные значения показателей надежности для системы в целом или ее подсистем;

2) перечень аварийных ситуаций, по которым должны быть регламентированы требования к надежности, и значения соответствующих показателей;

3) требования к надежности технических средств и программного обеспечения;

4) требования к методам оценки и контроля показателей надежности на разных стадиях создания системы в соответствии с действующими нормативно-техническими документами.

В требования по безопасности включают требования до обеспечению безопасности при монтаже, наладке, эксплуатации, обслуживании и ремонте технических средств системы (защита от воздействий электрического тока, электромагнитных полей, акустических шумов и т. п.), по допустимым уровням освещенности, вибрационных и шумовых нагрузок.

В требования по эргономике и технической эстетике включают показатели ИС, задающие необходимое качество взаимодействия человека с машиной и комфортность условий работы персонала.

Для подвижных ИС в требования к транспортабельности включают конструктивные требования, обеспечивающие транспортабельность технических средств системы, а также требования к транспортным средствам.

В требования к эксплуатации, техническому обслуживанию, ремонту и хранению включают:

1) условия и регламент (режим) эксплуатации, которые должны обеспечивать использование технических средств (ТС) системы с заданными техническими показателями, в том числе виды и периодичность обслуживания ТС системы или допустимость работы без обслуживания;

2) предварительные требования к допустимым площадям для размещения персонала и ТС системы, к параметрам сетей энергоснабжения и т. п.;

3) требования по количеству, квалификации обслуживающего персонала и режимам его работы;

4) требования к составу, размещению и условиям хранения комплекта запасных изделий и приборов;

5) требования к регламенту обслуживания.

В требования к защите информации от несанкционированного доступа включают требования, установленные в НТД, действующей в отрасли (ведомстве) заказчика.

В требованиях по сохранности информации приводят перечень событий: аварий, отказов технических средств (в том числе - потеря питания) и т. п., при которых должна быть обеспечена сохранность информации в системе.

В требованиях к средствам защиты от внешних воздействий приводят:

- 1) требования к радиоэлектронной защите средств ИС;
- 2) требования по стойкости, устойчивости и прочности к внешним воздействиям (среде применения).

В требованиях по патентной чистоте указывают перечень стран, в отношении которых должна быть обеспечена патентная чистота системы и ее частей.

В требования к стандартизации и унификации включают: показатели, устанавливающие требуемую степень использования стандартных, унифицированных методов реализации функций (задач) системы, поставляемых программных средств, типовых математических методов и моделей, типовых проектных решений, унифицированных форм управленческих документов, установленных ГОСТ 6.10.1, общесоюзных классификаторов технико-экономической информации и классификаторов других категорий в соответствии с областью их применения, требования к использованию типовых- автоматизированных рабочих мест, компонентов и комплексов.

В дополнительные требования включают:

- 1) требования к оснащению системы устройствами для обучения персонала (тренажерами, другими устройствами аналогичного назначения) и документацией на них;
- 2) требования к сервисной аппаратуре, стендам для проверки элементов системы;
- 3) требования к системе, связанные с особыми условиями эксплуатации;
- 4) специальные требования по усмотрению разработчика или заказчика системы.

В подразделе “Требование к функциям (задачам)”, выполняемым системой, приводят:

- 1) по каждой подсистеме перечень функций, задач или их комплексов (в том числе обеспечивающих взаимодействие частей системы), подлежащих автоматизации;

при создании системы в две или более очереди - перечень функциональных подсистем, отдельных функций или задач, вводимых в действие в 1-й и последующих очередях;

- 2) временной регламент реализации каждой функции, задачи (или комплекса задач);

3) требования к качеству реализации каждой функции (задачи или комплекса задач), к форме представления выходной информации, характеристики необходимой точности и времени выполнения, требования одновременности выполнения группы функций, достоверности выдачи результатов;

4) перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

В подразделе “Требования к видам обеспечения” в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другие видам обеспечения системы.

Для математического обеспечения системы приводят требования к составу, области применения (ограничения) и способам, использования в системе математических методов и моделей, типовых алгоритмов и алгоритмов, подлежащих разработке.

Для информационного обеспечения: системы приводят требования:

1) к составу, структуре и способам организации данных в системе;

2) к информационному обмену между компонентами системы;

3) к информационной совместимости со смежными системами;

4) по использованию общесоюзных и зарегистрированных республиканских, отраслевых классификаторов, унифицированных документов и классификаторов, действующих на данном предприятии;

5) по применению систем управления базами данных;

6) к структуре процесса сбора, обработки, передачи данных в системе и представлению данных;

7) к защите данных от разрушений при авариях и сбоях в электропитании системы;

8) к контролю, хранению, обновлению и- восстановлению данных;

9) к процедуре придания юридической силы документам, производимым техническими средствами ИС (в соответствии с ГОСТ 6.10.4).

Для лингвистического обеспечения системы приводят требования к применению в системе языков программирования высокого уровня, языков взаимодействия пользователей и технических

средств системы, а также требования к кодированию и декодированию данных, к языкам ввода-вывода данных, языкам манипулирования данными, средствам описания предметной области (объекта автоматизации), к способам организации диалога.

Для программного обеспечения системы приводят перечень покупных программных средств, а также требования:

- 1) к независимости программных средств от используемых СВТ и ОС;
- 2) к качеству ПС, а также к способам его обеспечения и контроля;
- 3) по необходимости согласования вновь разрабатываемых ПС с фондом алгоритмов и программ.

Для технического обеспечения системы приводят требования:

- 1) к видам технических средств, в том числе к видам комплексов технических средств, программно-технических комплексов и других комплектующих изделий, допустимых к использованию в системе;
- 2) к функциональным, конструктивным и эксплуатационным характеристикам средств технического обеспечения системы.

В требованиях к метрологическому обеспечению приводят:

- 1) предварительный перечень измерительных каналов;
- 2) требования к точности измерений параметров и (или) к метрологическим характеристикам измерительных каналов;
- 3) требования к метрологической совместимости технических средств системы;
- 4) перечень управляющих и вычислительных каналов системы, для которых необходимо оценивать точностные характеристики;
- 5) требования к метрологическому обеспечению технических и программных средств, входящих в состав измерительных каналов системы, средств, встроенного контроля, метрологической пригодности измерительных каналов и средств измерений, используемых при наладке и испытаниях системы;
- 6) вид метрологической аттестации (государственная или ведомственная) с указанием порядка ее выполнения и организаций, проводящих аттестацию.

Для организационного обеспечения приводят требования:

- 1) к структуре и функциям подразделений, участвующих в функционировании системы или обеспечивающих эксплуатацию;

- 2) к организации функционирования системы и порядку взаимодействия персонала ИС и персонала объекта автоматизации;
- 3) к защите от ошибочных действий персонала системы.

Для методического обеспечения приводят требования к составу нормативно-технической документации системы (перечень применяемых при ее функционировании стандартов, нормативов, методик и т. п.).

Раздел **“Состав и содержание работ по созданию (развитию) системы”** должен содержать перечень стадий и этапов работ по созданию системы в соответствии с ГОСТ 24.601, сроки их выполнения, перечень организаций — исполнителей работ, ссылки на документы, подтверждающие согласие этих организаций на участие в создании системы, или запись, определяющую ответственного (заказчик или разработчик) за проведение этих работ.

В данном разделе также приводят:

- 1) перечень документов, по ГОСТ 34.201, предъявляемых по окончании соответствующих стадий и этапов работ;
- 2) вид и порядок проведения экспертиза технической документации (стадия, этап, объем проверяемой документации, организация-эксперт);
- 3) программу работ, направленных на обеспечение требуемого уровня надежности разрабатываемой системы (при необходимости);
- 4) перечень работ по метрологическому обеспечению на всех стадиях создания системы с указанием их сроков выполнения и организаций-исполнителей (при необходимости).

В разделе **“Порядок контроля и приемки системы”** указывают:

- 1) виды, состав, объем и методы испытаний системы и ее составных частей (виды испытаний в соответствии с действующими нормами, распространяющимися на разрабатываемую систему);
- 2) общие требования к приемке работ по стадиям (перечень участвующих предприятий и организаций, место и сроки проведения), порядок согласования и утверждения приемочной документации;
- 3) статус приемочной комиссии (государственная, межведомственная, ведомственная).

В разделе **“Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие”**

необходимо привести перечень основных мероприятий и их исполнителей, которые следует выполнить при подготовке объекта автоматизации к вводу ИС в действие.

В перечень основных мероприятий включают:

- 1) приведение поступающей в систему информации (в соответствии с требованиями к информационному и лингвистическому обеспечению) к виду, пригодному для обработки с помощью ЭВМ;
- 2) изменения, которые необходимо осуществить в объекте автоматизации;
- 3) создание условий функционирования объекта автоматизации, при которых гарантируется соответствие создаваемой системы требованиям, содержащимся в ТЗ;
- 4) создание необходимых для функционирования системы подразделений и служб;
- 5) сроки и порядок комплектования штатов и обучения персонала.

Например, для АСУ приводят:

- изменения применяемых методов управления;
- создание условий для работы компонентов АСУ, при которых гарантируется соответствие системы требованиям, содержащимся в ТЗ.

В разделе **“Требования к документированию”** приводят:

- 1) согласованный разработчиком и Заказчиком системы перечень подлежащих разработке комплектов и видов документов, соответствующих требованиям ГОСТ 34.201 и НТД отрасли заказчика; перечень документов, выпускаемых на машинных носителях; требования к микрофильмированию документации;
- 2) требования по документированию комплектующих элементов межотраслевого применения в соответствии с требованиями ЕСКД и ЕСПД;
- 3) при отсутствии государственных стандартов, определяющих требования к документированию элементов системы, дополнительно включают требования к составу и содержанию таких документов.

В разделе **“Источники разработки”** должны быть перечислены документы и информационные материалы (технико-экономическое обоснование, отчеты о законченных научно-исследователь-

ских работах, информационные материалы на отечественные, зарубежные системы-аналоги и др.), на основании которых разрабатывалось ТЗ и которые должны быть использованы при создании системы.

В состав ТЗ на ИС при наличии утвержденных методик включают приложения, содержащие:

- 1) расчет ожидаемой эффективности системы;
- 2) оценку научно-технического уровня системы.

Приложения включают в состав ТЗ на ИС по согласованию между разработчиком и заказчиком системы.

3. ПРАВИЛА ОФОРМЛЕНИЯ

3.1. Разделы и подразделы ТЗ на ИС должны быть размещены в порядке, установленном в разд. 2 настоящего стандарта.

3.2. ТЗ на ИС оформляют в соответствии с требованиями ГОСТ 2.105 на листах формата А4 по ГОСТ 2.301 без рамки, основной надписи и дополнительных граф к ней.

Номера листов (страниц) проставляют, начиная с первого листа, следующего за титульным листом, в верхней части листа (над текстом, посередине) после обозначения кода ТЗ на ИС.

3.3. Значения показателей, норм и требований указывают, как правило, с предельными отклонениями или максимальным и минимальным значениями. Если эти показатели, нормы, требования однозначно регламентированы НТД, в ТЗ на ИС следует приводить ссылку на эти документы или их разделы, а также дополнительные требования, учитывающие особенности создаваемой системы. Если конкретные значения показателей, норм и требований не могут быть установлены в процессе разработки ТЗ на ИС, в нем следует сделать запись о порядке установления и согласования этих показателей, норм и требований:

“Окончательное требование (значение) уточняется в процессе „И согласовывается протоколом с ... на стадии ...”. При этом в текст ТЗ на ИС изменений не вносят.

3.4. На титульном листе помещают подписи заказчика, разработчика и согласующих организаций, которые скрепляют гербовой печатью. При необходимости титульный лист оформляют на не-

скольких страницах. Подписи разработчиков ТЗ на ИС и должностных лиц, участвующих в согласовании и рассмотрении проекта ТЗ на ИС, помещают на последнем листе.

Форма титульного листа ТЗ на ИС приведена в приложении 2. Форма последнего листа ТЗ на ИС приведена в приложении 3.

3.5. При необходимости на титульном листе ТЗ на ИС допускается помещать установленные в отрасли коды, например: гриф секретности, код работы, регистрационный номер ТЗ и др.

3.6. Титульный лист дополнения к ТЗ на ИС оформляют аналогично титульному листу технического задания. Вместо наименования “Техническое задание” пишут “Дополнение № ... к ТЗ на АС ...”.

3.7. На последующих листах дополнения, к ТЗ на ИС помещают основание для изменения, содержание изменения и ссылки на документы, в соответствии с которыми вносятся эти изменения.

3.8. При изложении текста дополнения к ТЗ следует указывать номера соответствующих пунктов, подпунктов, таблиц основного ТЗ на ИС и т. п. и применять слова: “заменить”, “дополнить”, “исключить”, “изложить в новой, редакции”.

3. Задание на практическую работу

Разработать ТЗ на проектирование ИС согласно варианту задания.

4 Контрольные вопросы

1. Какие разделы и подразделы содержит ТЗ?
2. Какой ГОСТ определяет требования ТЗ?
3. Какие данные размещаются на титульном листе ТЗ?
4. Как в ТЗ определяется порядок приемки и контроля?
5. Что такое организационное обеспечение проекта?
6. Что такое техническое обеспечение проекта?
7. Что такое метрологическое обеспечение проекта?
8. Что указывается в требованиях к программному обеспечению проекта?
9. Как формулируются функциональные требования к системе?
10. Какие правила и сроки согласования ТЗ?

Список литературы

1. Вендров, А. М. Проектирование программного обеспечения экономических информационных систем [Текст] : учебник для студ. вуз. / А. М. Вендров. - М. : Финансы и статистика, 2000. - 352 с.
2. Карпова, Т. С. Базы данных [Электронный ресурс] : модели, разработка, реализация / Т. С. Карпова. - Москва : Интернет-Университет Информационных Технологий, 2008. - 357 с.
3. Смирнова, Г. Н. Проектирование экономических информационных систем [Текст] : учебник / А. А. Сорокин, Ю. Ф. Тельнов. - М. : Финансы и статистика, 2003.
4. Романов, В. П. Проектирование экономических информационных систем. Методология и современные технологии [Текст] : учебное пособие / В. П. Романов, Н. З. Емельянова, Т. Л. Партыка ; Российская экономическая академия им. Г. В. Плеханова. - М. : Экзамен, 2005. - 256 с.
5. Проектирование информационных систем [Электронный ресурс] : методические указания по изучению дисциплины для студентов и бакалавров специальностей 351400 (0808062.62) «Прикладная информатика (в экономике)» / Юго-Зап. гос. ун-т ; сост. В. Н. Николаев. - Курск : ЮЗГУ, 2010. - 22 с

Порядок разработки согласования и утверждения ТЗ на проектирование ИС

1. Проект ТЗ на ИС разрабатывает организация-разработчик системы с участием заказчика на основании технических требований' (заявки; тактико-технического задания и т. п.). При конкурсной организации работ варианты проекта ТЗ на ИС рассматриваются заказчиком, который - либо выбирает предпочтительный, вариант, либо на основании сопоставительного анализа подготавливает с участием будущего разработчика ИС окончательный вариант ТЗ на ИС.

2. Необходимость согласования проекта ТЗ на ИС с органами государственного надзора и другими заинтересованными организациями определяют совместно заказчик системы и разработчик проекта ТЗ на ИС. Работу по согласованию проекта ТЗ на ИС осуществляют совместно разработчик ТЗ на ИС и заказчик системы, каждый в организациях своего министерства (ведомства).

3. Срок согласования проекта ТЗ на ИС в каждой организации не должен превышать 15 дней со дня его получения. Рекомендуется рассылать на согласование экземпляры проекта ТЗ на ИС (копии) одновременно во все организации (подразделения).

4. Замечания по проекту ТЗ на ИС должны быть представлены с техническим обоснованием. Решения по замечаниям должны быть приняты разработчиком проекта ТЗ на ИС и заказчиком системы до утверждения ТЗ на ИС.

5. Если при согласовании проекта ТЗ на ИС возникли разногласия между разработчиком и заказчиком (или другими заинтересованными организациями), то составляется протокол разногласий (форма произвольная) и конкретное решение принимается в установленном порядке.

6. Согласование проекта ТЗ на ИС разрешается оформлять отдельным документом (письмом). В, этом случае под грифом "Согласовано" делают ссылку на этот документ.

7. Утверждение ТЗ на ИС осуществляют руководители предприятий (организаций) разработчика и заказчика системы.

8. ТЗ на ИС (дополнение к ТЗ) до передачи его на утверждение должно быть проверено службой нормоконтроля организации - разработчика ТЗ и, при необходимости, подвергнуто метрологической экспертизе.

&. Копии, утвержденного ТЗ на ИС в 10-дневный срок после утверждения высылаются разработчиком ТЗ на ИС участникам создания системы.

10. Согласование и утверждение дополнений к ТЗ на ИС проводят в порядке, установленном для ТЗ на ИС.

11. Изменения к ТЗ на ИС не допускается утверждать после представления системы для ее очереди на приемо-сдаточные испытания.

12. Регистрация, учет и хранение ТЗ на ИС и дополнений к нему проводят в соответствии, с требованиями ГОСТ 2.501.

Практическая работа №3

Планирование разработки ИС на основе диаграммы Ганта

1. Цель работы

Разработать диаграмму Ганта выполнения проекта

2. Порядок разработки диаграммы Ганта

Диаграммы Ганта – это способ графически изобразить ход проекта, порядок выполнения задач, их продолжительность, время начала и завершения. Рассмотрим на примере использование диаграммы Ганта.

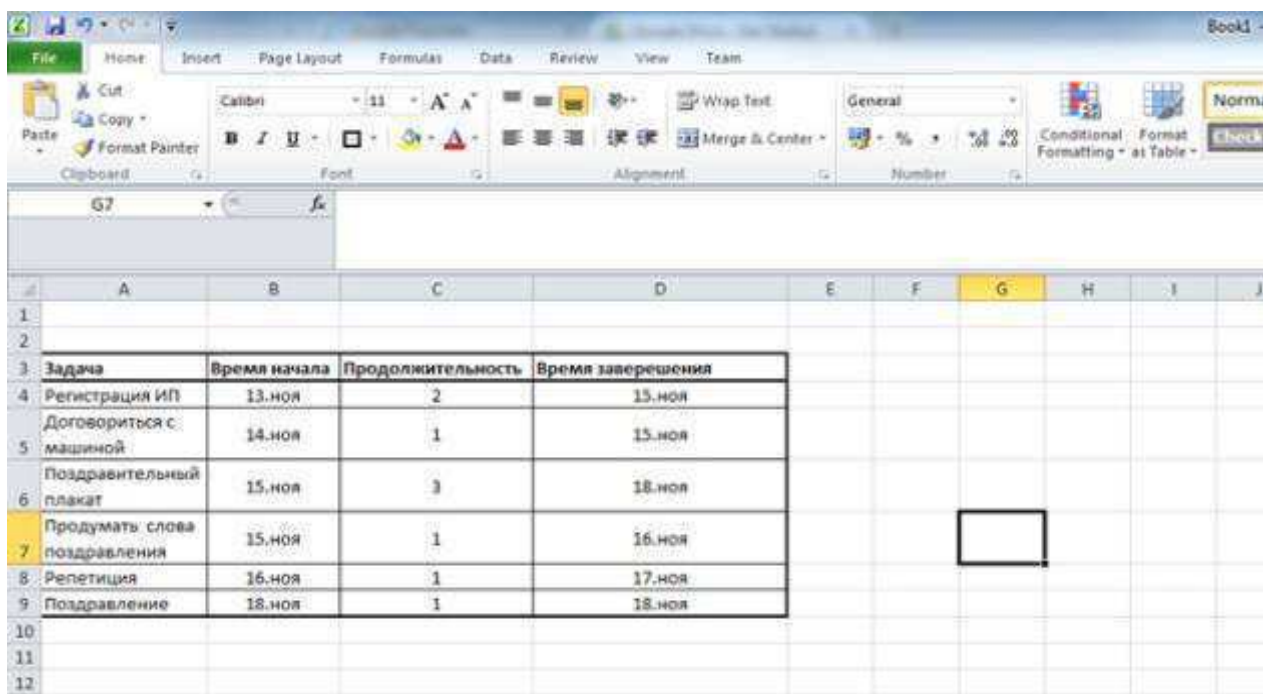
Таблица 4 – Перечень этапов выполнения проекта ИС

Этап проекта	Начало	Длительность	Конец
Изучение и анализ области внедрения	20.11.2016	3	22.11.2016
Составление технического задания	23.11.2016	4	26.11.2016
Планирование и контроль проектных работ.	Планирование и контроль проектных работ.	Планирование и контроль проектных работ.	Контроль проектных работ.
Описание входных и выходных данных	02.12.2016	7	08.12.2016
Разработка структур данных	09.12.2016	8	16.12.2016
Разработка технического проекта	17.12.2016	10	26.12.2016
Написание программ, модулей утилит	27.12.2016	10	05.01.2017
Отладка	06.01.2017	7	12.01.2017
Тестирование	13.01.2017	5	17.01.2017
Разработка справочной и технической документации	18.01.2017	2	19.01.2017
Внедрение	20.01.2017	7	26.01.2017
Итого		71	дней

Определим, как нам можно грамотно распланировать задачи с помощью диаграммы Ганта.

Воспользуемся инструментом Microsoft Excel. Построим диаграмму Ганта с помощью Microsoft Excel 2010.

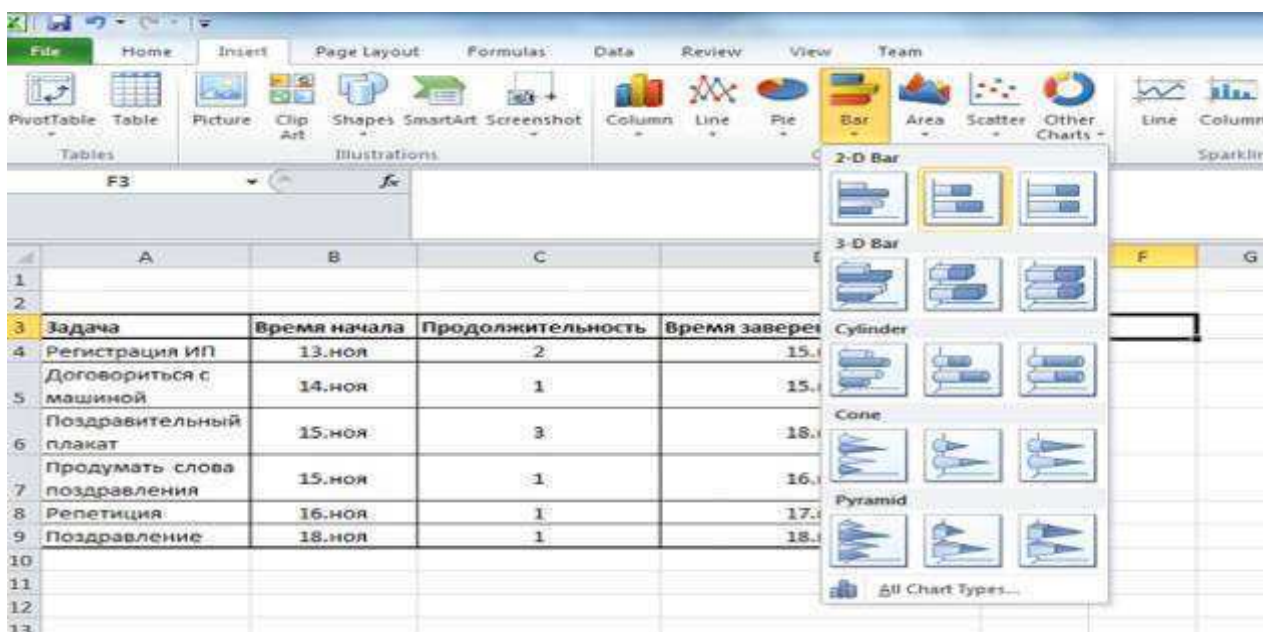
Открываем инструмент Excel и переносим наши задачи.



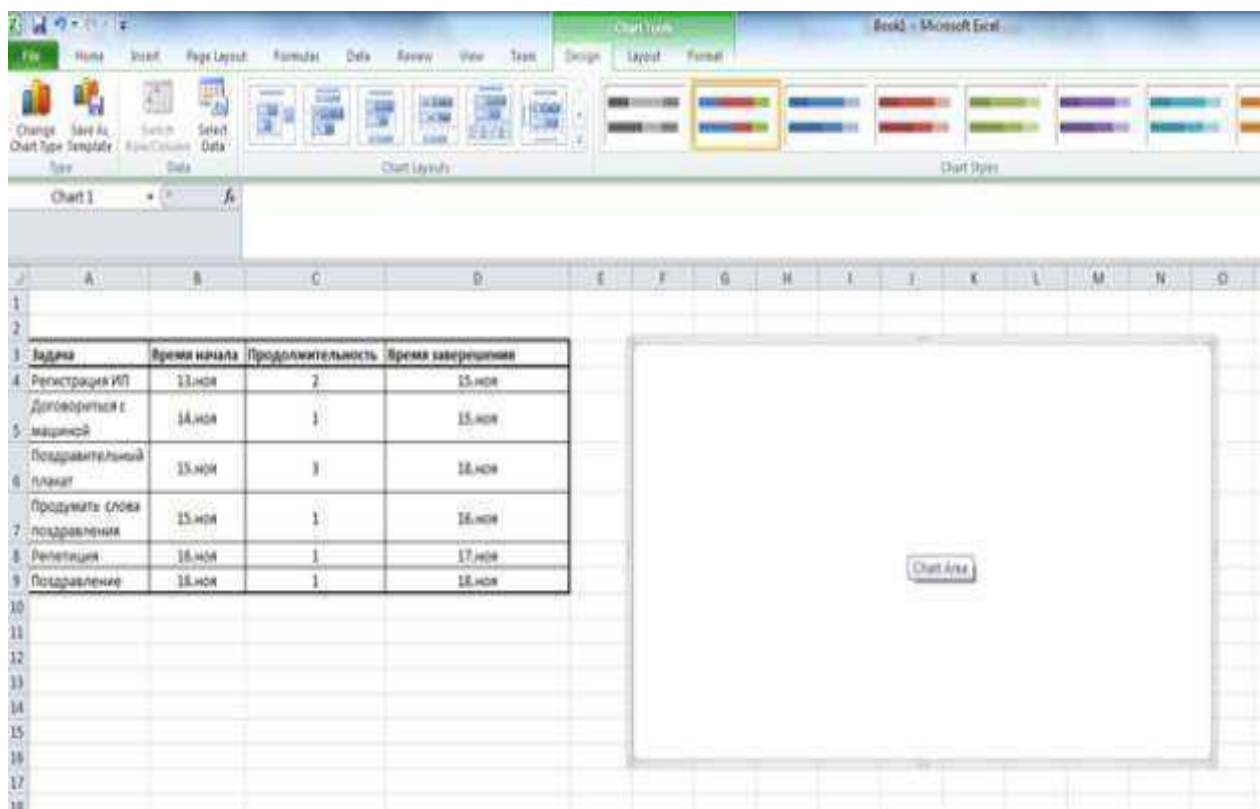
The screenshot shows the Microsoft Excel 2010 interface with a table of tasks. The table has four columns: 'Задача' (Task), 'Время начала' (Start Time), 'Продолжительность' (Duration), and 'Время завершения' (End Time). The tasks are listed in rows 4 through 9.

Задача	Время начала	Продолжительность	Время завершения
Регистрация ИП	13.ноя	2	15.ноя
Договориться с машиной	14.ноя	1	15.ноя
Поздравительный плакат	15.ноя	3	18.ноя
Продумать слова поздравления	15.ноя	1	16.ноя
Репетиция	16.ноя	1	17.ноя
Поздравление	18.ноя	1	18.ноя

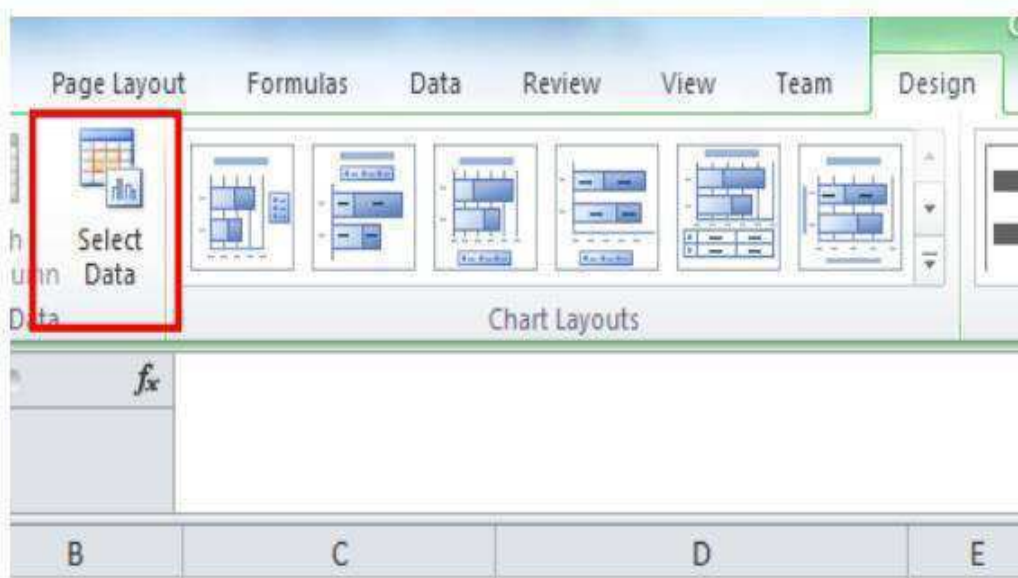
Следующая вещь, которую нам нужно сделать – это выбрать Insert -> Bar. И выбрать тот список диаграмм, который мы хотим отобразить



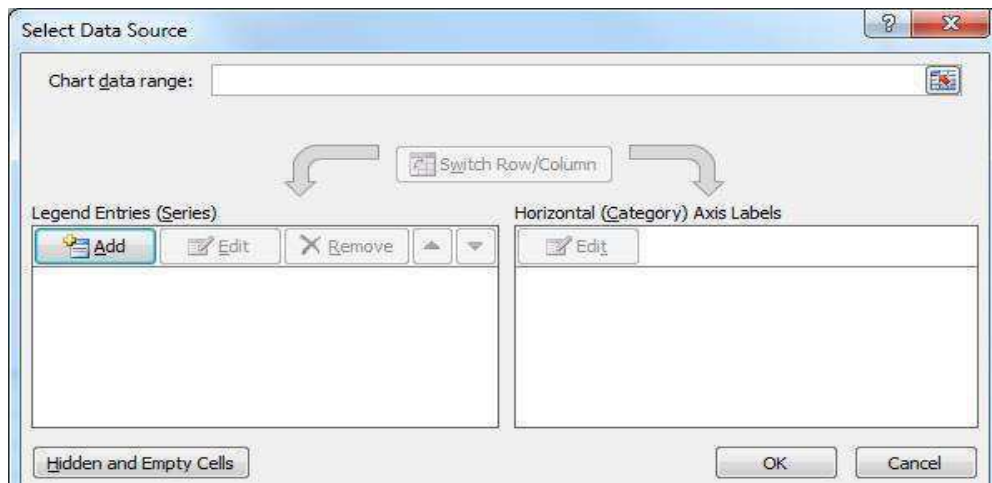
Появившуюся область мы отображаем в том месте, где хотим, чтобы наш график отображался. И теперь нам нужно связать эту область с нашими данными.



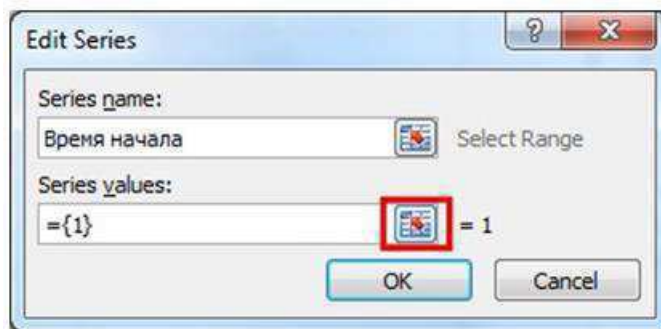
Далее на вкладке Design нажимаем кнопку Select Data.



В появившемся окне нажимает кнопку Add.

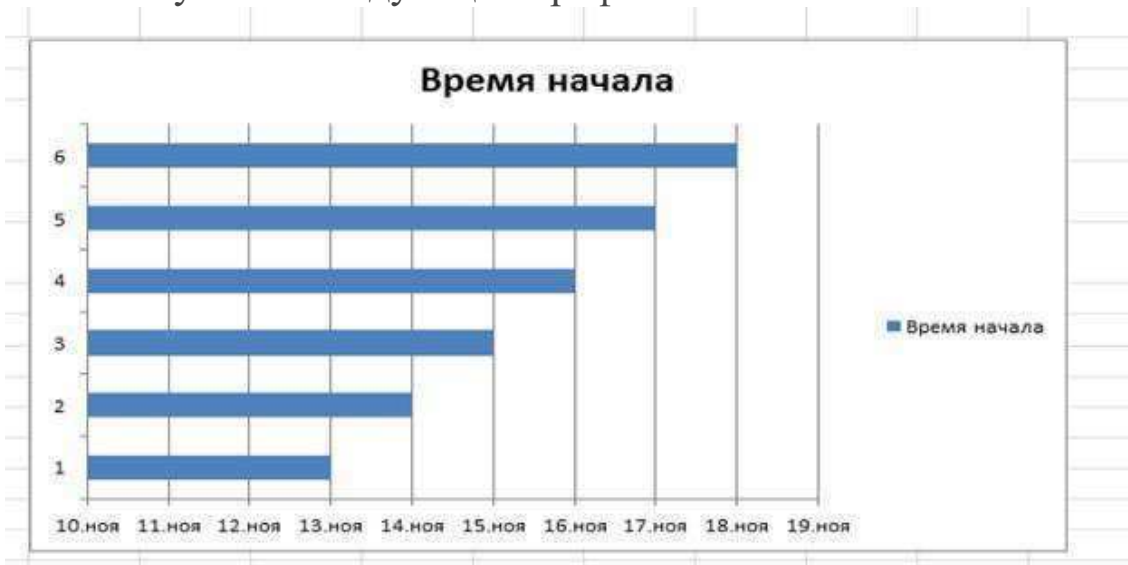


В появившемся окне, там где series name вводим Время начала и нажимаем на кнопку выделенную красным.

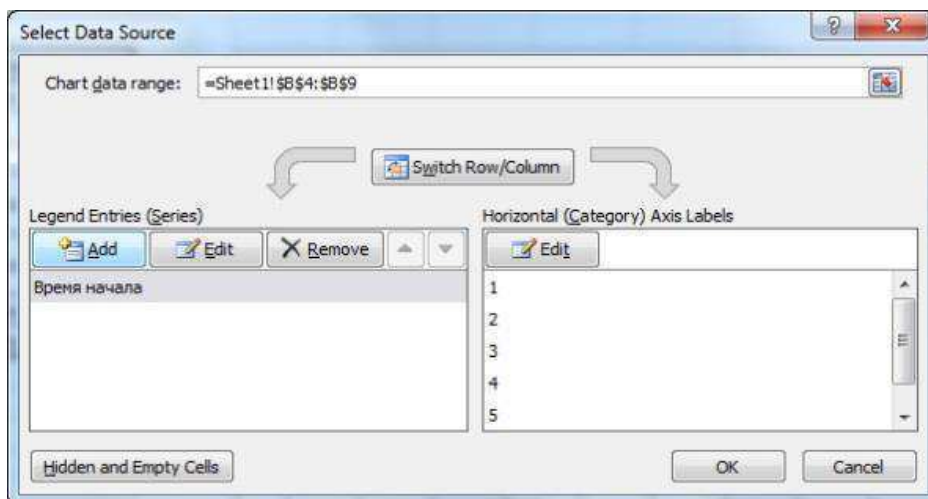


Когда появится окно с одним полем нажимаем на поле в окне Edit Series, после чего выделяем промежуток из таблицы, нужный нам и нажимаем на кнопку, выделенную красным.

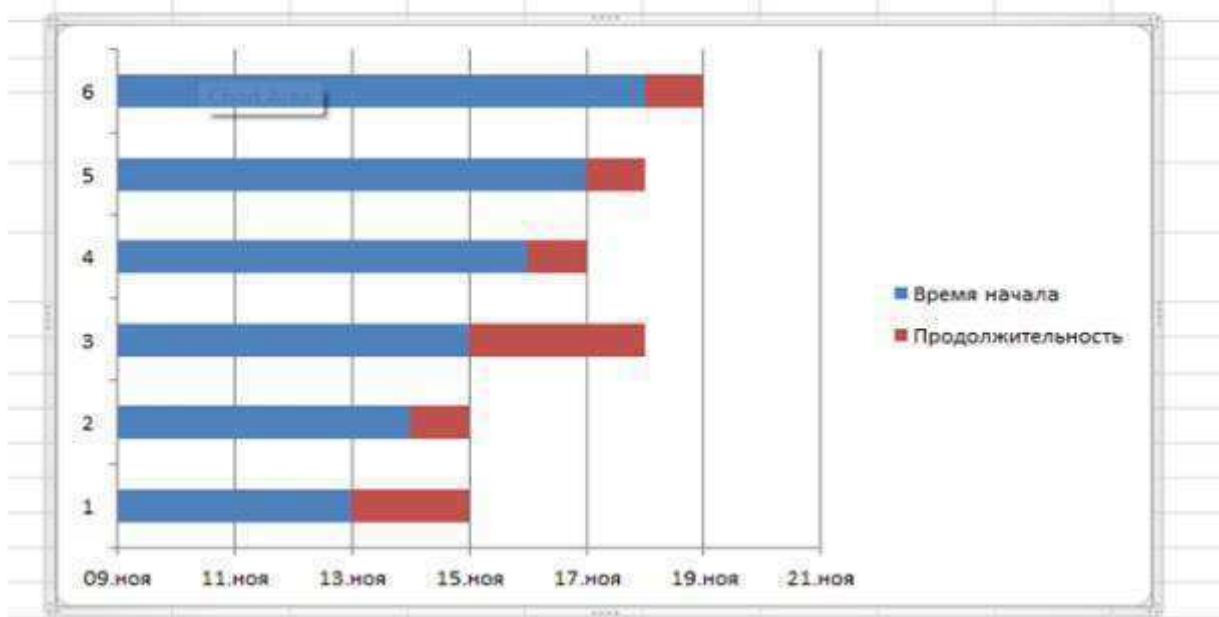
Получился следующий график



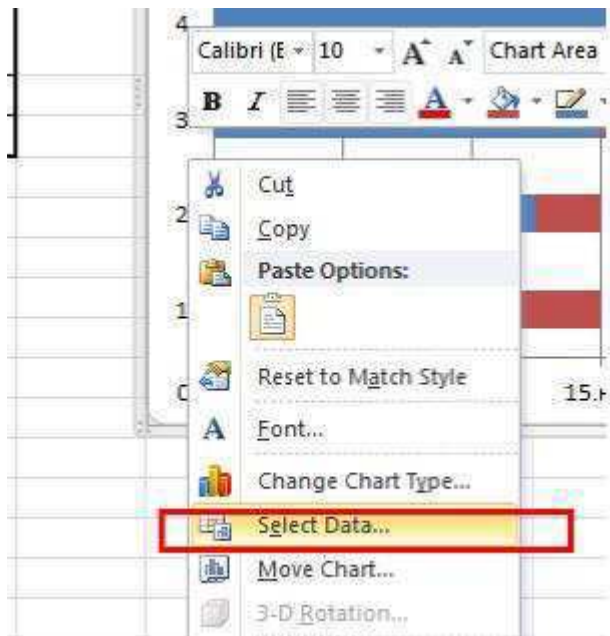
Необходимо в график добавить данные по продолжительности. Для этого в появившемся окне снова нажимаем кнопку Add и выбрать значения из графы продолжительность.



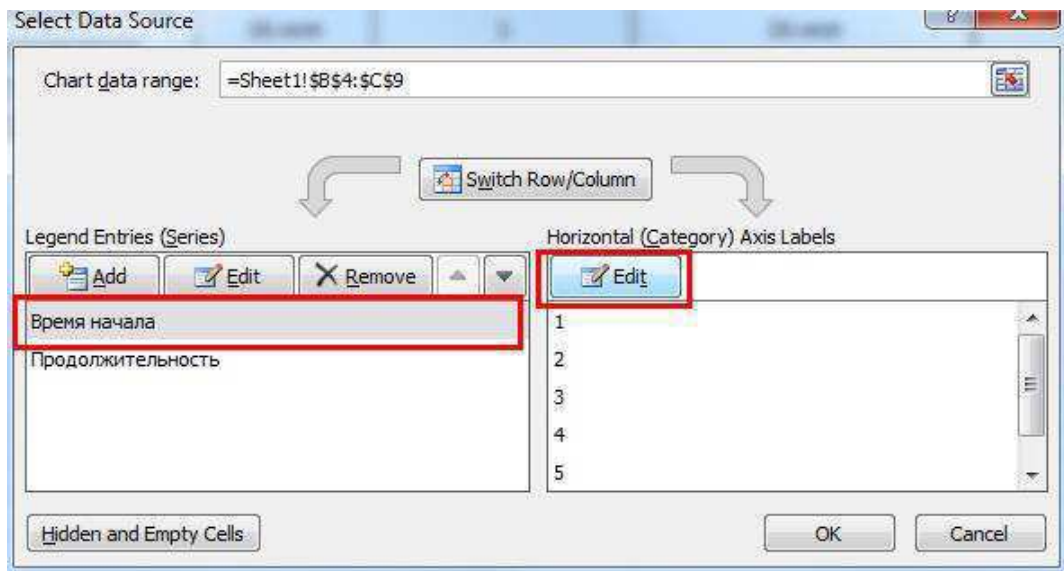
В результате у нас получился вот такой вот график.



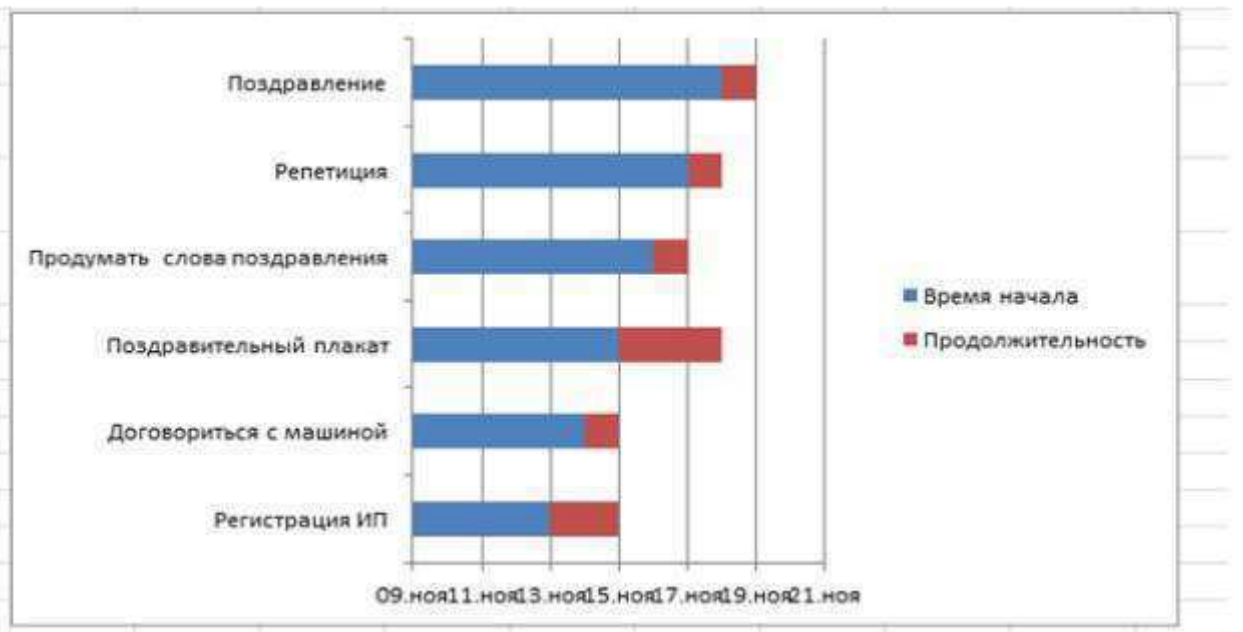
Следующее, что мы сделаем – это заменим цифры, напротив колонок на названия из нашей таблицы (Столбик “Задача”). Для этого я щелкаю правой кнопкой мыши по цифрам и в появившемся меню выбираю Select Data.



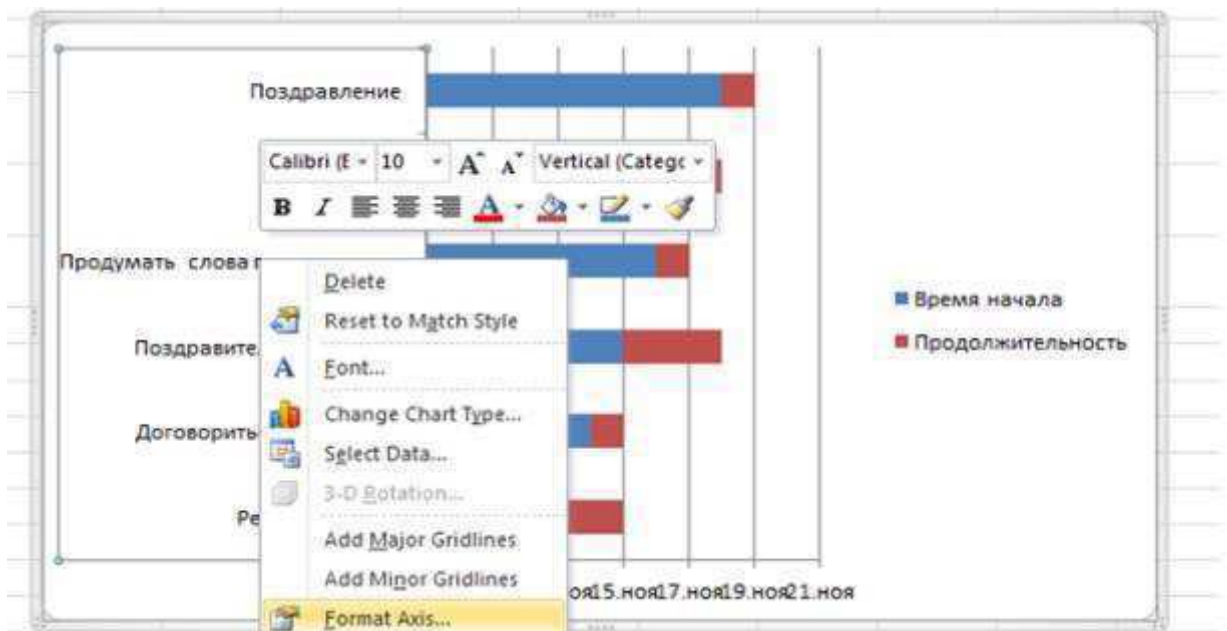
Появляется окно, в котором нужно выделить время начала и нажать кнопку Edit (выделено красным).



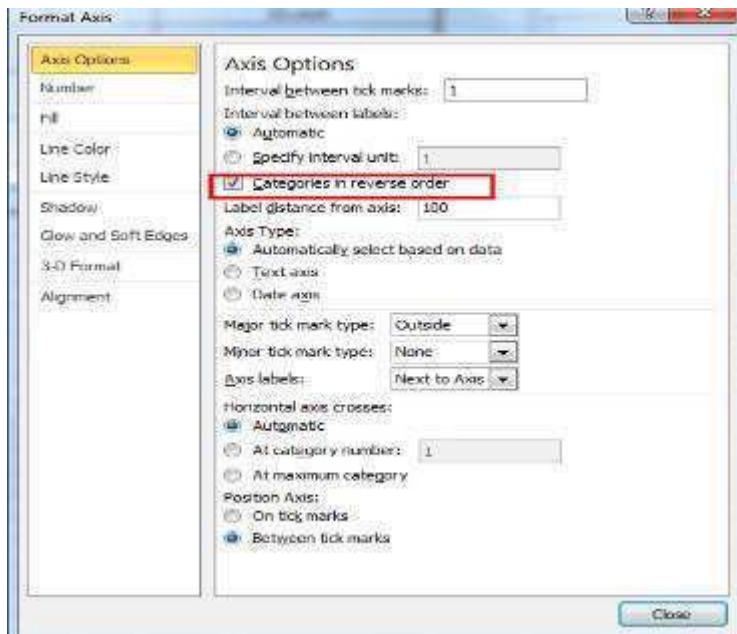
После чего выделить промежуток, откуда планируется брать данные (по аналогии, как это делалось выше) и нажать на кнопку Ok. Как мы видим цифры заменились словами



Задачи отображаются в обратном порядке снизу вверх. Для изменения порядка кликаем правой кнопкой мыши по словам. И выбираем опцию **Format Axis**.



И в появившемся окне ставим галочку в строке выделенной красным.

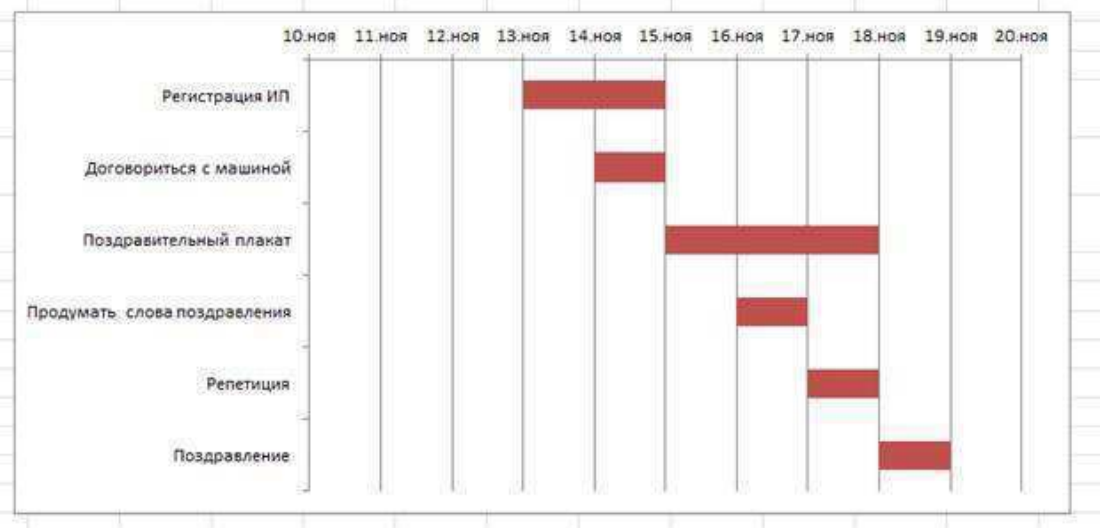


Следующее, что нужно сделать – это спрятать синие колонки.

Для этого выделяем их, после чего кликаем по ним правой клавишей и выбираем опцию **Format Data Series**

В появившемся окне сначала выбираем **Fill->No fill**, потом **Border Color ->No line**. Диаграмма приобрела следующий вид

Для того, чтобы убрать фразы **Время начала** и **Продолжительность** и расширить поле с графиком выбираем и удаляем их. Получается вот такой вот график.



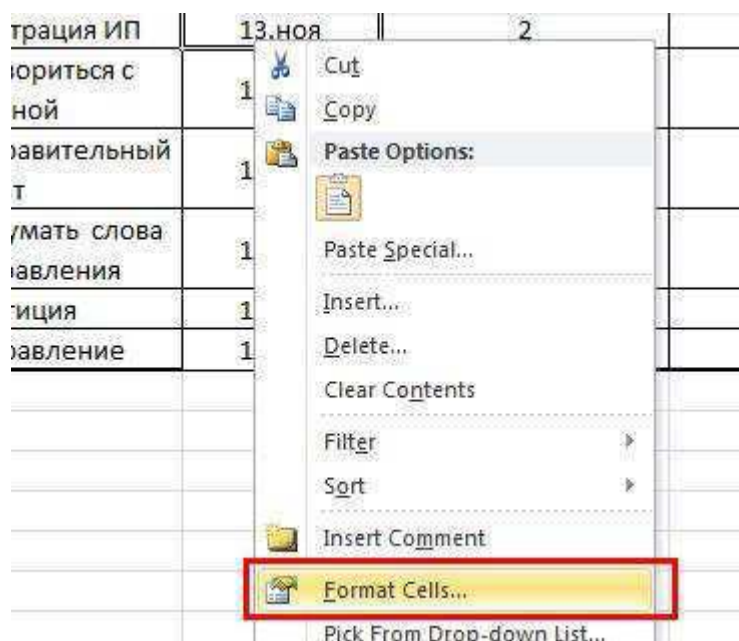
Последнее, что можно сделать – это сдвинуть весь наш график влево, так что отсчет начиналась с 13 ноября.

Для это выделяем поле с датами, щелкаем на него правой кнопкой мыши и выбираем опцию **Format Axis**.

В появившемся окне в поле Minimum выбираем Fixed и меняем дату (ее числовое значение) на ту, которая необходима нам.

Как понять, какую дату нужно вписывать в это поле?

Для этого можно щелкнуть правой кнопкой мыши на поле с той датой, которая нам необходима и выбрать опцию Format Cells



Нажать General и в правом углу вы увидите нужную вам дату.

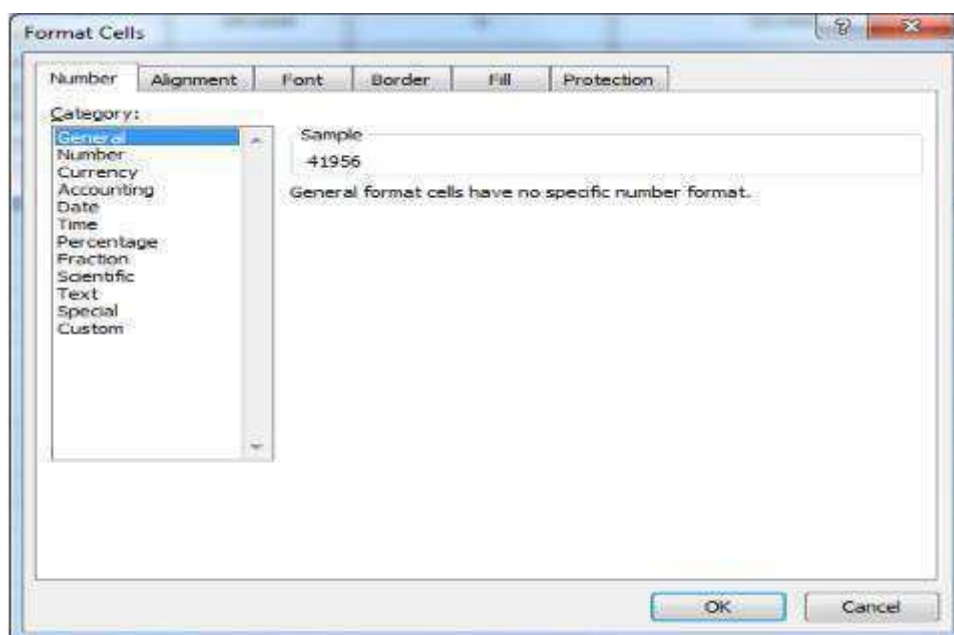


Диаграмма Ганта с помощью Microsoft Excel 2010 построена.

3. Задание к лабораторной работе

Разработать: диаграмму Ганта для реализации проекта разработки информационной системы выбранной предметной области.

Контрольные вопросы

1. Какие этапы жизненного цикла ИС рассматриваются при составлении графика работ по проекту ИС?
2. С какой целью строится график выполнения работ по проекту?
3. Кто определяет порядок выполнения задач, их продолжительность, время начала и завершения?
4. Каким образом производится оценка срока выполнения проекта в целом?
5. Каким инструментальным средством можно разработать график выполнения проекта ИС?
6. Как создать на диаграмме работу и определить начало и продолжительность работы?
7. Как выбрать вид диаграммы Гантта?
8. Можно ли поменять порядок выполнения работ проекта?
9. От чего зависит срок выполнения работ проекта?
10. Можно ли изменить время реализации проекта?

Список литературы

1. Лапина, Т. И. Информационные системы. Проектный практикум к выполнению и защите ВКР бакалавра по направлению 09.03.02 Информационные системы, 09.03.03 Прикладная информатика/ Т. И. Лапина//Юго-Западный гос. ун-т, ЗАО «Университетская книга»–Курск, 2016.–99с.
2. Золотов, С. Ю. Проектирование информационных систем [Электронный ресурс] : учебное пособие / С. Ю. Золотов. - Томск : Эль Контент, 2013. - 88 с.
3. Абрамов, Г. В. Проектирование информационных систем [Электронный ресурс] : учебное пособие / Г. В. Абрамов, И. Медведкова, Л. Коробова. - Воронеж : Воронежский государственный университет инженерных технологий, 2012. - 172 с.
4. Стасышин, В. М. Проектирование информационных систем и баз данных [Электронный ресурс] : учебное пособие / В. М. Стасышин. - Новосибирск : НГТУ, 2012. - 100 с.
5. Вендров, А.М. Проектирование программного обеспечения [Текст] : учебник / А.М. Вендров.— М: Финансы и Статистика, 2006. — 352с.

Практическая работа №4

Анализ сроков и ресурсов проекта с использованием сетевого планирования

1. Цель работы

Получить навыки описания и анализа сроков и ресурсов выполнения проекта информационной системы с использованием сетевого планирования.

2. Основные теоретические положения

Сетевое планирование включает в себя несколько этапов:

1. Разбиение проекта на совокупность работ, выполнение которых необходимо для реализации проекта;
2. Построение сетевого графика, описывающего последовательность выполнения работ;
3. Оценка временных характеристик работ и анализ сетевого графика.

Сетевой график – это ориентированный граф, в котором вершинами обозначены работы проекта, а дугами – временные взаимосвязи работ.

Сетевой график должен удовлетворять следующим **свойствам**:

1. Каждой работе соответствует одна вершина.
2. Ни одна работа не может быть представлена на сетевом графике дважды
3. Ни одна работа не может быть начата до того, как закончатся все непосредственно предшествующие ей работы, то есть, если в некоторую вершину входят дуги, то работа может начаться только после окончания всех работ, из которых выходят эти дуги.
4. Если из работы выходит несколько дуг, то ни одна из работ, в которые входят эти дуги, не может начаться до окончания текущей работы.

5. Начало и конец проекта обозначаются работами с нулевой продолжительностью.

Пример. В качестве примера рассмотрим проект "Разработка программного комплекса". Предположим, что проект состоит из работ, характеристики которых приведены в табл.1.

Таблица 1 – Перечень работ по проекту

Номер работы	Название работы	Длительность
1	Начало реализации проекта	0
2	Постановка задачи	10
3	Разработка программного интерфейса	5
4	Разработка модулей обработки данных	7
5	Разработка структуры базы данных	6
6	Заполнение базы данных	8
7	Отладка программного комплекса	5
8	Тестирование и исправление ошибок	10
9	Составление программной документации	5
10	Завершение проекта	0

Сетевой график для данного проекта изображен на рис.1. На нем вершины, соответствующие обычным работам, обведены тонкой линией, а толстой линией обведены вехи проекта.

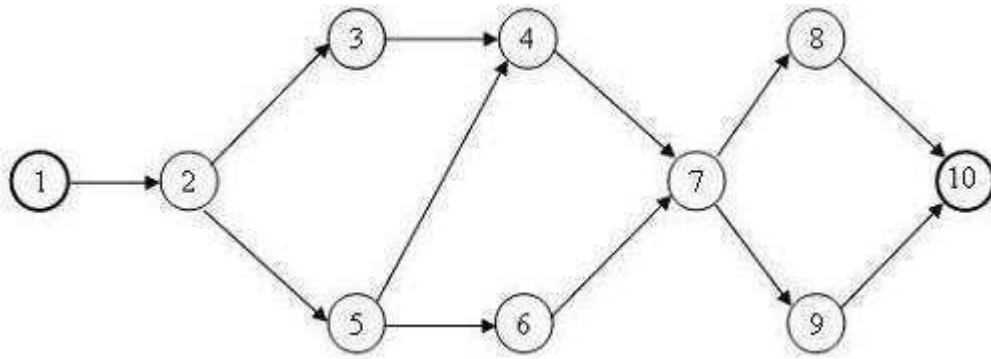


Рисунок 1 – Сетевой график проекта

Сетевой график позволяет по заданным значениям длительностей работ найти критические работы проекта и его критический путь.

Критической называется работа, для которой задержка ее начала приведет к задержке срока окончания проекта в целом.

Такие работы не имеют запаса времени. Некритические работы имеют некоторый запас времени, и в пределах этого запаса их начало может быть задержано.

Критический путь – это путь от начальной к конечной вершине сетевого графика, проходящий только через критические работы. Суммарная длительность работ критического пути определяет *минимальное время реализации проекта*.

Нахождение критического пути сводится к нахождению критических работ и выполняется в два этапа.

1. Вычисление **раннего времени начала** каждой работы проекта. Эта величина показывает время, раньше которого работа не может быть начата.

2. Вычисление **позднего времени начала** каждой работы проекта. Эта величина показывает время, позже которого работа не может быть начата без увеличения продолжительности всего проекта.

Критические работы имеют одинаковое значение раннего и позднего времени начала.

Обозначим:

t_i – время выполнения работы i ,

$T_p(i)$ – раннее время начала работы i ,

$T_n(i)$ – позднее время начала работы i .

Тогда

$$T_p(i) = \max (T_p(j) + t_j), j \in G$$

где G – множество работ, непосредственно предшествующих работе i .

Раннее время начальной работы проекта принимается равным нулю

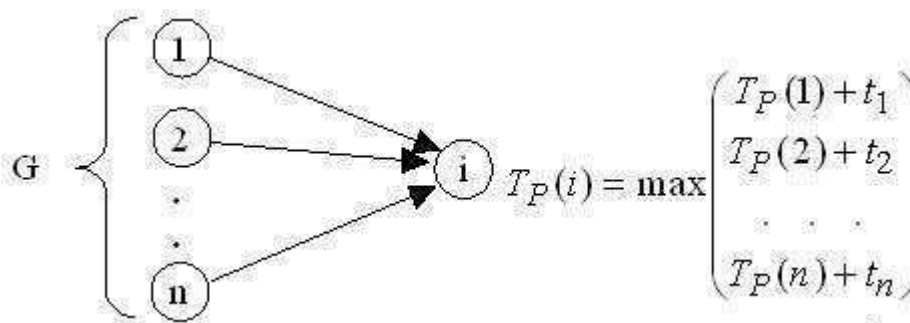


Рисунок 2 – Схема вычисления раннего времени начала работы

Поскольку последняя работа проекта – это веха нулевой длительности, раннее время ее начала совпадает с длительностью всего проекта.

Обозначим T – длительностью всего проекта.

Тогда, за T принимается позднее время начала последней работы.

Для остальных работ позднее время начала вычисляется по формуле:

$$T_n(i) = \min (T_n(j) - t_i) , j \in H$$

Где H – множество работ, непосредственно следующих за работой i .

Схематично вычисления раннего и позднего времени начала изображены, соответственно, на рис.2 и рис.3.

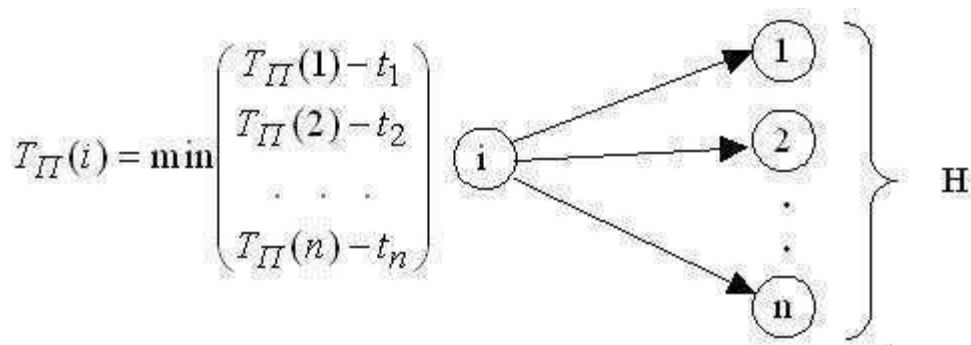


Рисунок 3 – Схема вычисления позднего времени начала работы

Пример:

Найдем критические работы и критический путь для проекта "Разработка программного комплекса", сетевой график которого изображен на рис.1, а длительности работ исчисляются днями и заданы в табл.1.

Сначала вычисляем раннее время начала каждой работы. Вычисления начинаются от начальной и заканчиваются конечной работой проекта.

Процесс и результаты вычислений изображены на рис.4.

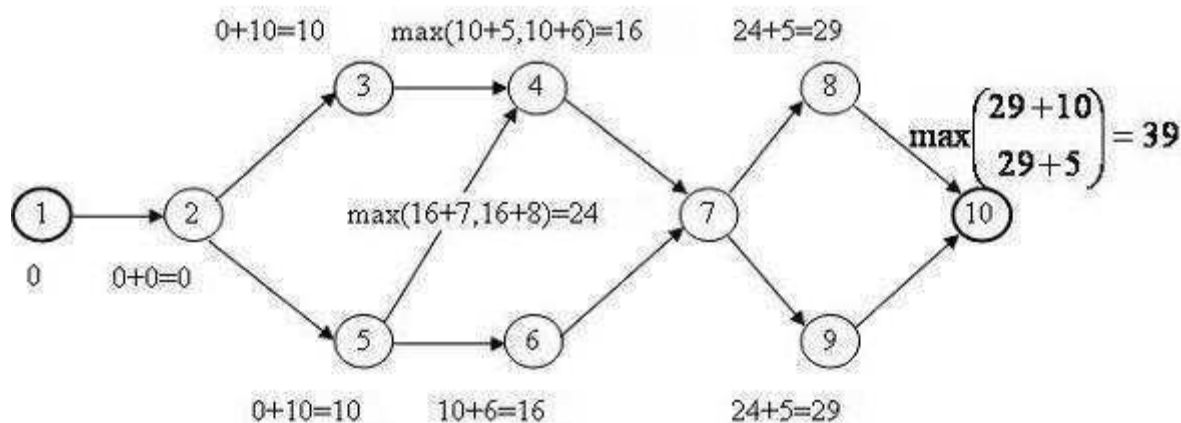


Рисунок 4 – Вычисление раннего времени начала работ

Т.к. начало проекта имеет 0 длительность, то

$$T_p(2) = \max(T_p(0) + 0) = 0$$

$$T_p(3) = 0 + 10 \text{ (длительность 2 работы)}$$

$$T_p(5) = 0 + 10 \text{ (длительность 2 работы)}$$

Из работы 5 выходят 2 стрелки, тогда по правилу, что если в некоторую вершину входят дуги, то работа может начаться только

после окончания всех работ, из которых выходят эти дуги, раннее время начала работы 4 определяется:

$$T_p(4) = \max(10+5 \text{ (длительность 3 работы)}, 10+6 \text{ (длительность 5 работы)}) = 16$$

В работу 7 входят 2 стрелки, тогда по такому же правилу раннее время начала работы 7 определяется:

$$T_p(7) = \max(16+7 \text{ (длительность 4 работы)}, 16+8 \text{ (длительность 6 работы)}) = 24$$

Из работы 7 выходят две стрелки, тогда по правилу если из работы выходит несколько дуг, то ни одна из работ, в которые входят эти дуги, не может начаться до окончания текущей работы, раннее время начала

$$T_p(8) = 24+5 \text{ (длительность 8 работы)} = 29$$

$$T_p(9) = 24+5 \text{ (длительность 9 работы)} = 29$$

В работу 10 входят две стрелки, тогда по правилу если в некоторую вершину входят дуги, то работа может начаться только после окончания всех работ, из которых выходят эти дуги, раннее время начала работы 10 определяется:

$$T_p(10) = \max(29+10 \text{ (длительность 8 работы)}, 29+5 \text{ (длительность 9 работы)}) = 39$$

Таким образом, результатом первого этапа помимо раннего времени начала работ является общая длительность проекта **T=39 дней**.

На следующем этапе вычисляем позднее время начала работ. Вычисления начинаются в последней и заканчиваются в первой работе проекта. Процесс и результаты вычислений изображены на рис.5.

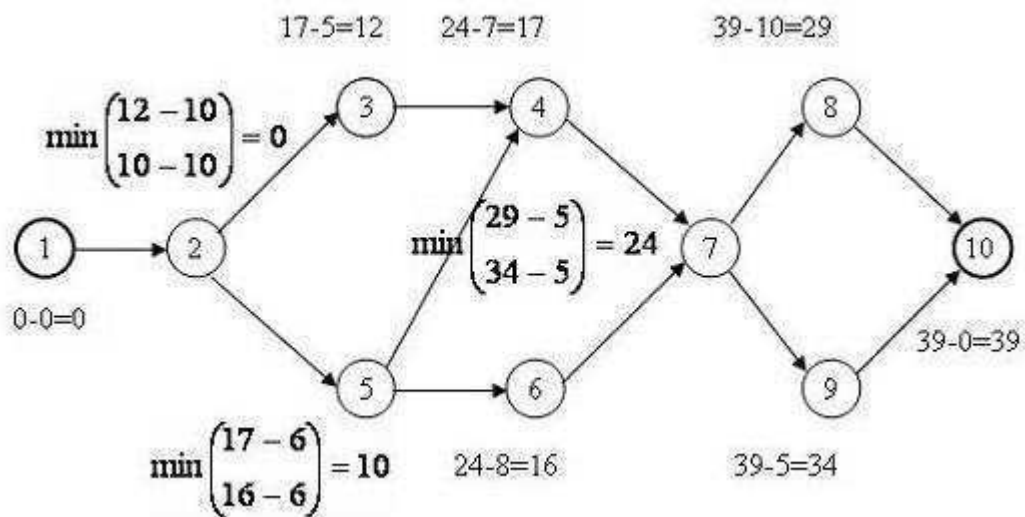


Рисунок 5 – Вычисление позднего времени начала работ

Сводные результаты расчетов приведены в табл.2.

В ней выделены заливкой критические работы.

Критический путь получается соединением критических работ на сетевом графике.

Он показан пунктирными стрелками на рис.6.

Таблица 2 – Сводные результаты работы

Работа	1	2	3	4	5	6	7	8	9	10
Раннее время начала	0	0	10	16	10	16	24	29	29	39
Позднее время начала	0	0	12	17	10	16	24	29	34	39
Резерв времени	0	0	2	1	0	0	0	0	5	0

После вычисления величин $T_p(i)$ и $T_n(i)$ для каждой работы вычисляется **резерв времени** $R(i)$:

$$R(i) = T_n(i) - T_p(i)$$

Эта величина показывает, насколько можно задержать начало работы i без увеличения длительности всего проекта.

Для критических работ резерв времени равен нулю. Поэтому усилия менеджера проекта должны быть направлены в первую очередь на обеспечение своевременного выполнения этих работ.

Для некритических работ резерв времени больше нуля, что дает менеджеру возможность маневрировать временем их начала и используемыми ими ресурсами.

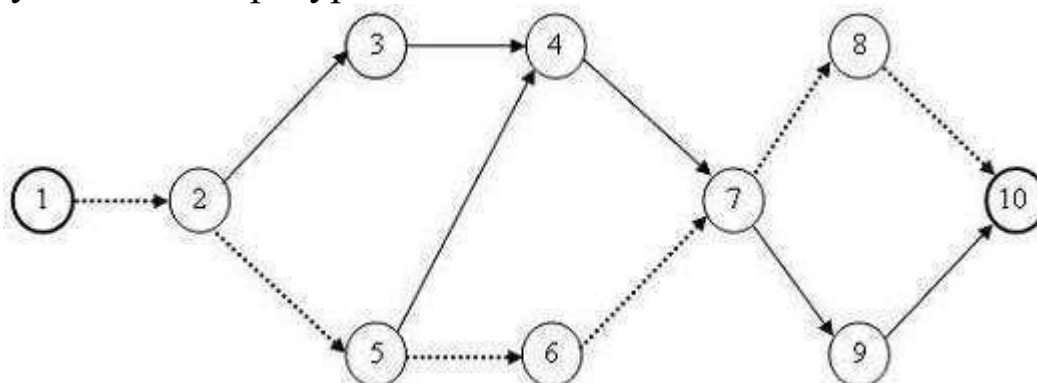


Рисунок 6 – Критический путь проекта

Анализ времени выполнения проекта на основе сетевого графика называют прямым (расчет раннего времени) и обратным (расчет позднего времени).

После проведения расчетов проводится анализ выполнения проекта.

Возможны такие варианты.

1. Задержка начала работы на величину, не превышающую резерв времени, а требуемые для работы ресурсы направляются для выполнения работ критического пути. Это может дать уменьшение длительности критической работы и проекта в целом.

2. Недогрузка некритической работы ресурсами. В результате длительность ее увеличивается в пределах резерва времени, а освободившийся ресурс задействуется для выполнения критической работы, что также приведет к уменьшению длительности ее и всего проекта.

Данные прямого и обратного анализа времени выполнения проекта на основе сетевого графика для наглядности оформляют следующим образом рис.7.

Такое представление позволяет анализировать ход выполнения и резервы времени.

3. Задание к лабораторной работе

Разработать: сетевой график реализации проекта разработки информационной системы выбранной предметной области.

Контрольные вопросы

1. С какой целью строится сетевой график выполнения работ по проекту?
2. Какие этапы включает процесс построения сетевого графика работ по проекту ИС?
3. Кто определяет порядок выполнения виды и порядок выполнения работ проекта, их продолжительность, время начала и завершения?
4. Каким образом производится оценка срока выполнения работ?
5. Каким образом определяются временные резервы работ?
6. Каким инструментальным средством можно разработать сетевой график выполнения проекта ИС?
7. Как создать на сетевой график и определить начало и продолжительность работы?
8. Что такое критический путь выполнения проекта?
9. Как определяются ранние и поздние сроки начала работ проекта?
10. От чего зависит срок выполнения работ проекта?

Список литературы

6. Лапина, Т. И. Информационные системы. Проектный практикум к выполнению и защите ВКР бакалавра по направлению 09.03.02 Информационные системы, 09.03.03 Прикладная информатика/ Т. И. Лапина//Юго-Западный гос. ун-т, ЗАО «Университетская книга»–Курск, 2016.–99с.
7. Золотов, С. Ю. Проектирование информационных систем [Электронный ресурс] : учебное пособие / С. Ю. Золотов. - Томск : Эль Контент, 2013. - 88 с.
8. Абрамов, Г. В. Проектирование информационных систем [Электронный ресурс] : учебное пособие / Г. В. Абрамов, И. Медведкова, Л. Коробова. - Воронеж : Воронежский государственный университет инженерных технологий, 2012. - 172 с.
9. Стасышин, В. М. Проектирование информационных систем и баз данных [Электронный ресурс] : учебное пособие / В. М. Стасышин. - Новосибирск : НГТУ, 2012. - 100 с.
10. Вендров, А.М. Проектирование программного обеспечения [Текст] : учебник / А.М. Вендров.— М: Финансы и Статистика, 2006. — 352с.

Практическая работа №5

Стоимостный анализ проекта ИС

1. В диалоге /ModelProperties(вызывается из меню Model/ModelProperties) во вкладке ABCUnits установите единицы измерения денег и времени — рубли и часы.

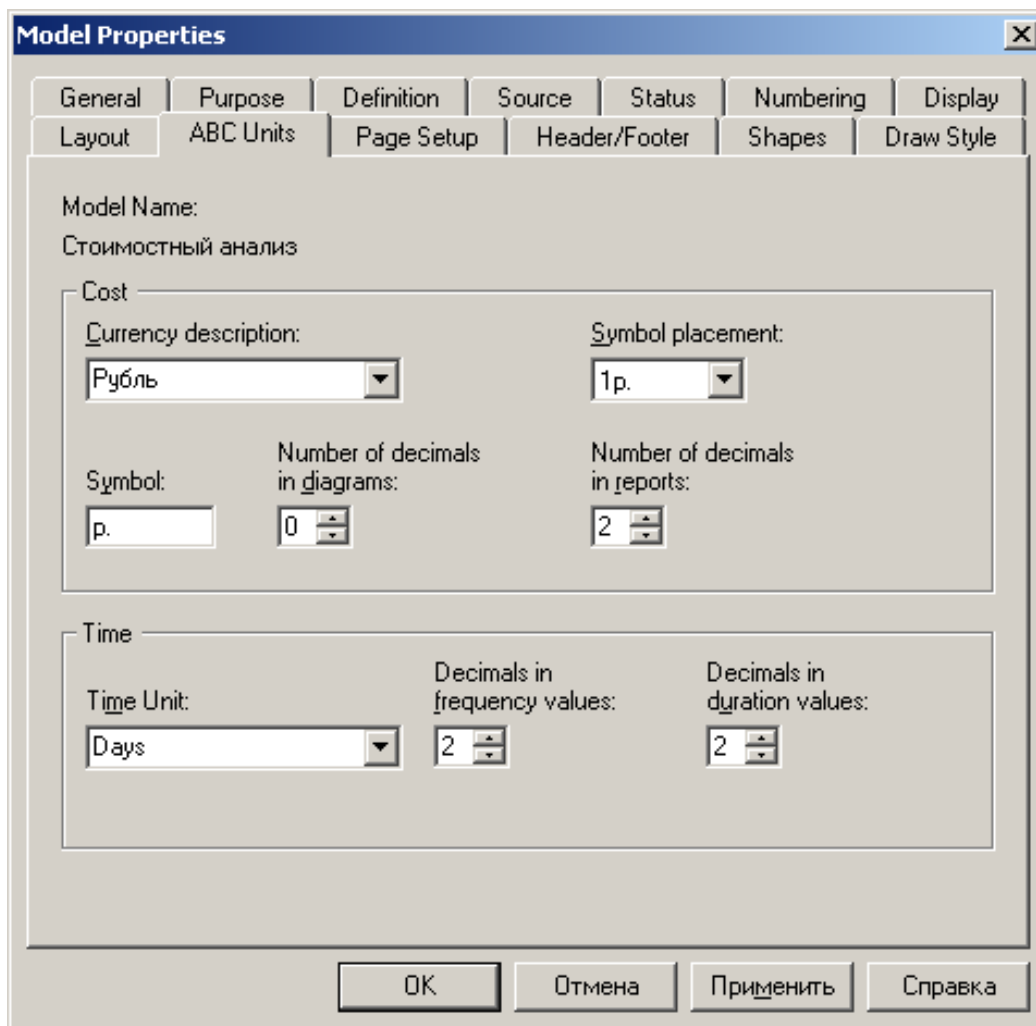


Рисунок 1 – Вкладка ABC Units диалога Model Properties

2. Перейдите в Dictionary/CostCenter и в диалоге CostCenterDictionary внесите название и определение центров затрат.

Таблица 1 – Определение центра затрат

Центр затрат	Определение
Управление	Затраты на управление, связанные с составлением графика работ, формированием партий компьютеров, контролем над сборкой и тестированием
Рабочая сила	Затраты на оплату рабочих, занятых сборкой и тестированием компьютеров
Компоненты	Затраты на закупку компонентов

Для отображения стоимости каждой работы в нижнем левом углу прямоугольника перейдите в меню Model/Model Properties и во вкладку Display диалого Model Properties включите опцию ABC Data (рис.2).

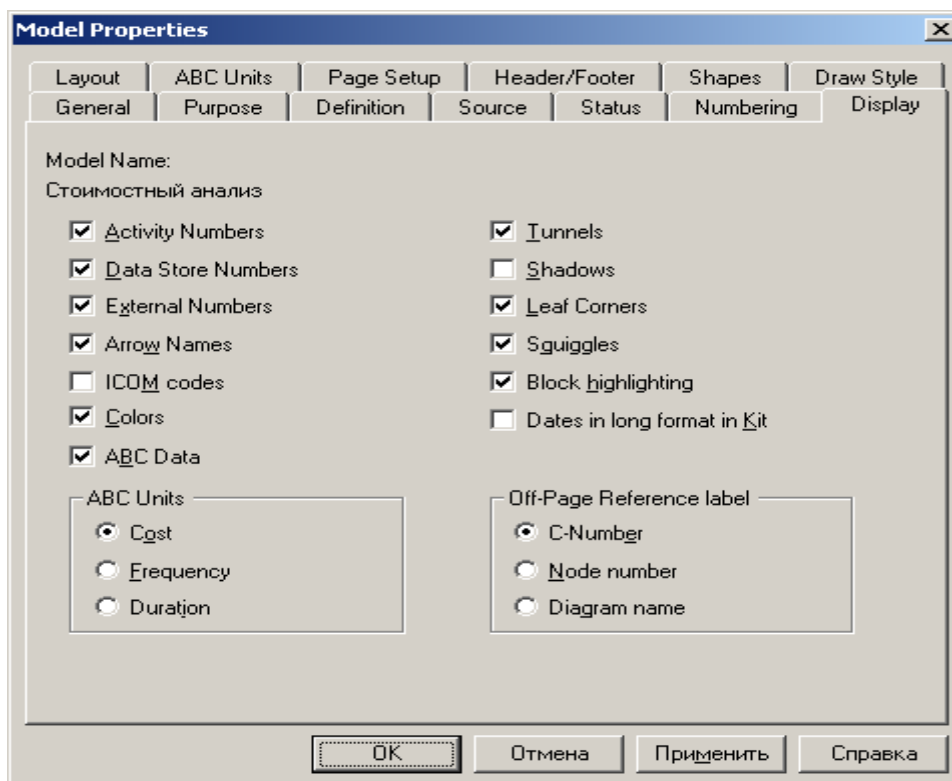


Рисунок 2 – Вкладка Display диалого Model Properties

Для отображения частоты или продолжительности работы переключите радиокнопки в группе ABCUnits.

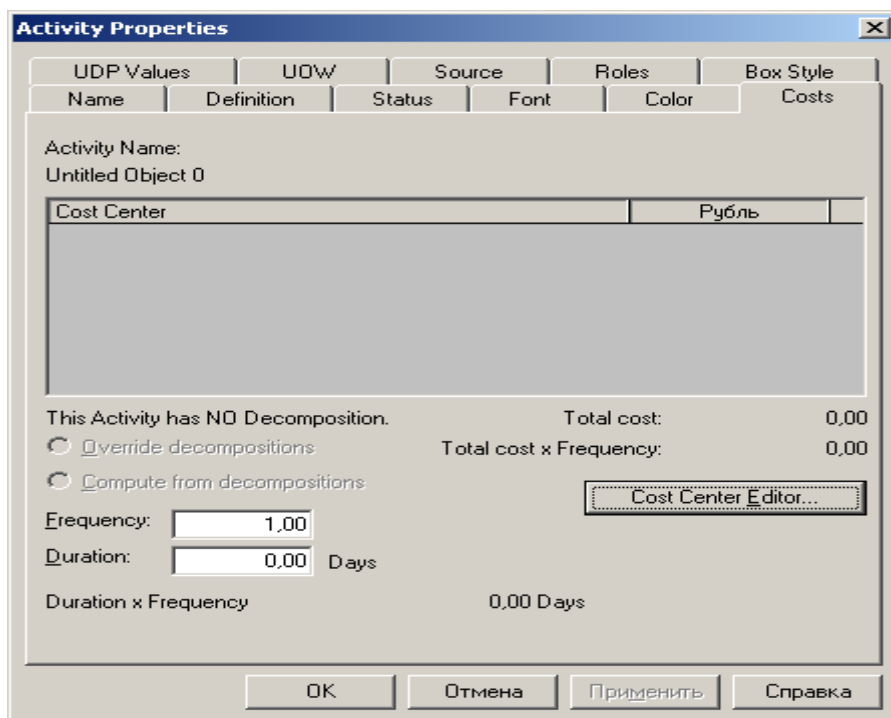


Рисунок 3 – Вкладка Cost диалога Activity Properties

Для назначения стоимости работе следует щелкнуть по ней правой кнопкой мыши и выбрать в контекстном меню Cost.

3. Для работ на диаграмме A2 внесите параметры ABC.

Таблица 2– Стоимости работ на диаграмме A2

Имя работы (Activity name)	Центр затрат (Cost Center)	Сумма центра затрат (Cost, руб.)	Продолжительность (Duration), день	Частота (Frequency)
Отслеживание расписания и управление сборкой и тестированием	Управление	500,00	1,00	1,00
Сборка настольных компьютеров	Рабочая сила	100,00	1,00	12,00
	Компоненты	16000,00		
		0		

Таблица 3 – Стоимости работ на диаграмме А2

Имя работы (Activity name)	Центр затрат (Cost Center)	Сумма центра затрат (Cost Center Cost),руб.	Продолжительность (Duration), день	Частота(Frequency)
Сборка ноутбуков	Рабочая сила	140,00	1,00	20,00
	Компоненты	28000,00		
Тестирование компьютеров	Рабочая сила	60,00	1,00	32,00

Посмотрите результат - стоимость работы верхнего уровня.

4. Контрольные вопросы

1. Какие вопросы включает методика описания проектируемой ИС?
2. Какие существуют документы для описания ИС?
3. На каких уровнях проводится обследование стоимости разработки?
4. Как определяется сумма затрат на разработку проекта?
5. Какие существуют универсальные методы, пригодные для расчета стоимостных затрат?
6. Какие статьи входят в расчет стоимости затрат?
7. Какие инструментальные средства используются при расчете стоимости затрат?
8. Какие виды работ оцениваются при расчете стоимости затрат?
9. Оцениваются ли материальные ресурсы для реализации проекта?
10. Цель анализа полученной информации о стоимости разработки системы?

Список литературы

1. Лапина, Т. И. Информационные системы. Проектный практикум к выполнению и защите ВКР бакалавра по направлению 09.03.02 Информационные системы, 09.03.03 Прикладная информатика/ Т. И. Лапина//Юго-Западный гос. ун-т, ЗАО «Университетская книга»–Курск, 2016.–99с.
2. Золотов, С. Ю. Проектирование информационных систем [Электронный ресурс] : учебное пособие / С. Ю. Золотов. - Томск : Эль Контент, 2013. - 88 с.
3. Абрамов, Г. В. Проектирование информационных систем [Электронный ресурс] : учебное пособие / Г. В. Абрамов, И. Медведкова, Л. Коробова. - Воронеж : Воронежский государственный университет инженерных технологий, 2012. - 172 с.
4. Стасышин, В. М. Проектирование информационных систем и баз данных [Электронный ресурс] : учебное пособие / В. М. Стасышин. - Новосибирск : НГТУ, 2012. - 100 с.
5. Вендров, А.М. Проектирование программного обеспечения [Текст] : учебник / А.М. Вендров.— М: Финансы и Статистика, 2006. — 352с.

Практическая работа №6

Разработка программного интерфейса проекта ИС

1. Цель работы

Получить практические навыки создания программных интерфейсов ИС при помощи Visual Studio и элементов управления Windows - формы.

2. Теоретические сведения

Разрабатываемое приложение предназначено для работы с базой данных сотрудников компании. На рис. 35 представлена структура базы данных.

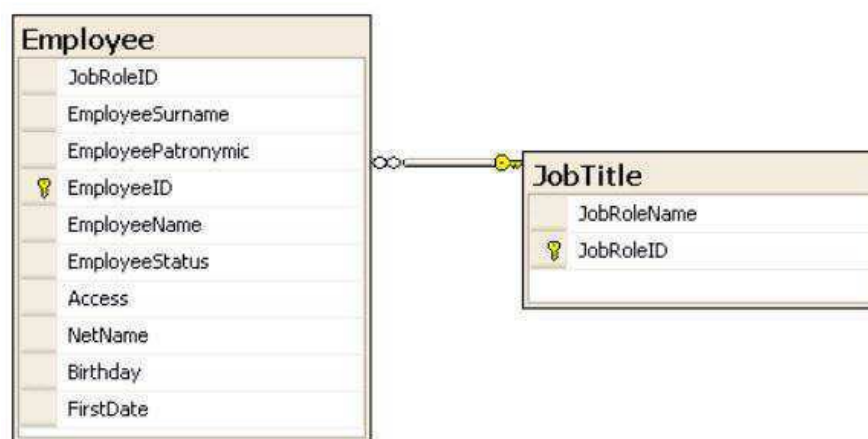


Рисунок 35 – Структура базы данных по сотрудникам компании

Создайте базу данных, включающую две таблицы:

сведения о сотрудниках - *Employee*;
справочник должностей - *JobTitle*.

Задайте значение атрибутам таблицы *Employee* приведенным в табл. 2

Таблица 2 – Атрибуты таблицы Employee

Имя атрибута	Назначение	Тип
<i>EmployeeID</i>	Суррогатный ключ	smallint
<i>JobRoleID</i>	Внешний ключ	smallint
<i>EmployeeSurname</i>	Фамилия	varchar(50)
<i>EmployeeName</i>	Имя	varchar(20)
<i>EmployeePatronymic</i>	Отчество	varchar(20)
<i>EmployeeStatus</i>	Статус	int
<i>Access</i>	Уровень доступа	varchar(20)
<i>NetName</i>	Сетевое имя	varchar(20)
<i>Birthday</i>	Дата рождения	Smalldatetime
<i>FirstDate</i>	Дата приема на работу	smalldatetime

Суррогатный ключ *EmployeeID*, как и все остальные суррогатные ключи базы данных, генерируется сервером базы данных автоматически, т.е. для него задано свойство `IDENTITY` для СУБД *MS SQL Server* или `AutoNumber` для *MS Access*. Атрибут *JobRoleID* является внешним ключом, с помощью которого осуществляется связь с табл. й *JobTitle*.

Назначение атрибутов таблицы *JobTitle* приведено в табл. 3.

Таблица 3 – Атрибуты таблицы JobTitle

Имя атрибута	Назначение	Тип
<i>JobRoleID</i>	Суррогатный ключ	smallint
<i>JobRoleName</i>	Наименование должности	varchar(50)

В рассматриваемом приложении в качестве СУБД используется *MS SQL Server*.

Создайте соединение проекта с базой данных. Для этого выберите пункт меню *Tools/Connect to Database*.

Появляется окно *AddConnection* (рис.36).

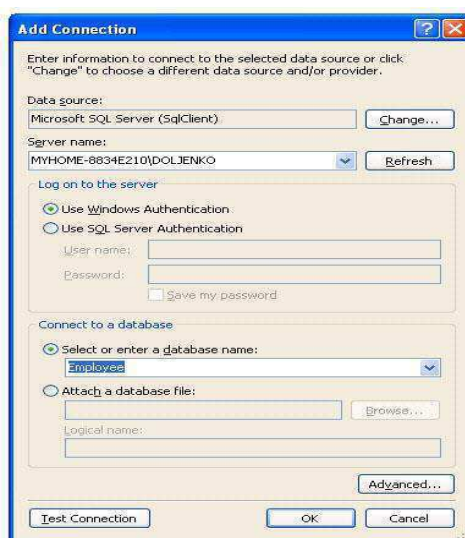


Рисунок 36 – Окно *AddConnection*

В пункте "*Server name*" задайте имя сервера, которое необходимо узнать у преподавателя. В пункте *Select or enter database name* - имя базы данных, которое определит преподаватель.

Для проверки правильности подключения к базе данных нажимаем клавишу "*Test Connection*". При правильном подключении появляется следующее сообщение (рис.37).



Рисунок 37 – Окно *Microsoft Data Link*

При нормальном соединении с базой данных можно открыть навигатор *Server Explorer* из меню *View/ Server Explorer* или сочетанием клавиш *ALT+CTRL+S* (рис.38).

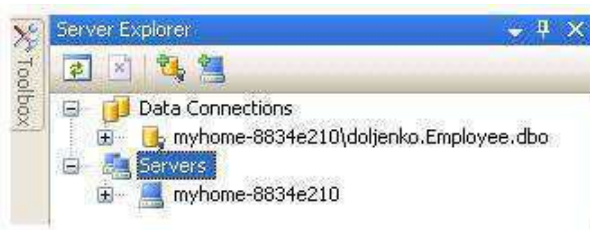


Рисунок 38 – Окно навигатора Server Explorer

Добавьте в проект объект класса DataSet. Для этого выберем пункт меню *Project/Add New Item . .* (рис.39).

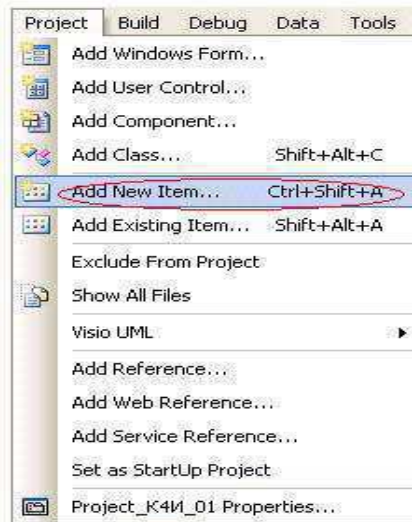


Рисунок 39 – Добавление в проект нового компонента

Выберите шаблон *DataSet* в окне *Add New Item* рис.40 и присвойте ему имя *DataSetEmployee*.



Рисунок 40 – Выбор нового компонента – DataSet

После нажатия кнопки *Add* система генерирует класс *DataSetEmployee*, который добавляется в решение проекта (рис.41).

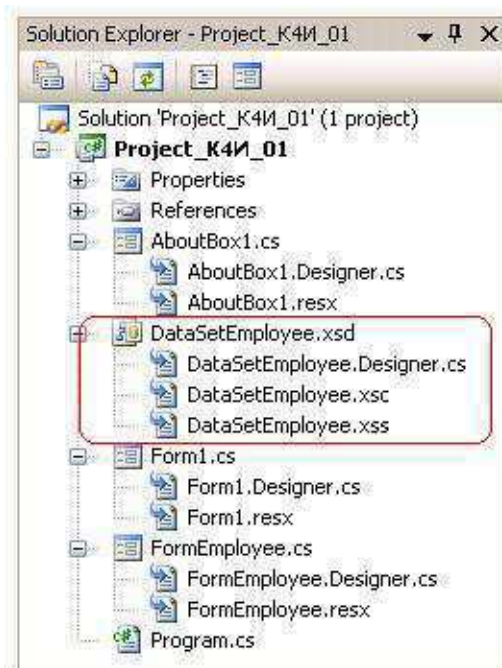


Рисунок 41 – Окно решения проекта с новым компонентом DataSet

Для добавления таблиц *Employee* и *JobTitle* к *DataSet* необходимо перетащить их из окна *Server Explorer* на поле графического дизайнера (рис.42).

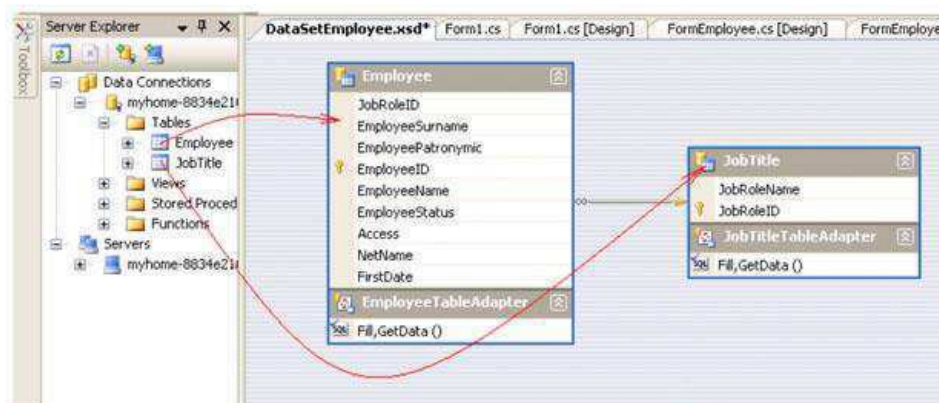


Рисунок 42 – Добавление таблиц к DataSet

В результате будут созданы классы таблиц, адаптеры и методы *Fill* и *GetData*.

При формировании класса `DataSetEmployee` необходимо учесть то, что первичные ключи таблиц *Employee* и *JobTitle* являются суррогатными и автоматически формируются (ключ со свойством автоинкремент) источником данных (например, *MS SQL Server*).

При формировании новых записей в приложении необходимо обеспечить уникальность первичных ключей для таблиц объекта `DataSetEmployee`. Это можно обеспечить, задав для ключевых колонок таблиц *Employee* и *JobTitle* следующие свойства:

```
AutoIncrement = true;  
AutoIncrementSeed = -1;  
AutoIncrementStep = -1;
```

Столбец со свойством `AutoIncrement` равным `true` генерирует последовательность значений, начинающуюся со значения `AutoIncrementSeed` и имеющую шаг `AutoIncrementStep`. Это позволяет генерировать уникальные значения целочисленного столбца первичного ключа. В этом случае при добавлении новой записи в таблицу будет генерироваться новое значение первичного ключа, начиная с -1, -2, -3 и т.д., которое никогда не совпадет с первичным ключом источника данных, т.к. в базе данных генерируются положительные первичные ключи. Свойства `AutoIncrementSeed` и `AutoIncrementStep` устанавливаются равными -1, чтобы гарантировать, что когда набор данных будет синхронизироваться с источником данных, эти значения не будут конфликтовать со значениями первичного ключа в источнике данных. При синхронизации `DataSet` с источником данных, когда добавляют новую строку в таблицу *MS SQL Server 2005* с первичным автоинкрементным ключом, значение, которое этот ключ имел в таблице *DataSet*, заменяется значением, сгенерированным СУБД.

Установите свойства `AutoIncrement`, `AutoIncrementSeed` и `AutoIncrementStep` для колонки первичного ключа `EmployeeID` таблицы *Employee* (рис.43).

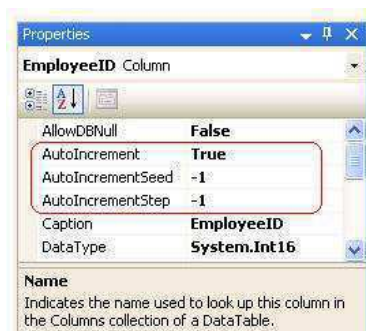


Рисунок 43 – Установка свойств для колонки EmployeeID

Сделайте для колонки JobTitleID таблицы JobTitle. установки свойств аналогичные AutoIncrement, AutoIncrementSeed и AutoIncrementStep

После создания класса DataSetEmployee и адаптера необходимо создать объекты этих классов, добавив следующий код к файлу FormEmployee.cs.

```
DataSetEmployee dsEmployee = new DataSetEmployee();
DataSetEmployeeTableAdapters.EmployeeTableAdapter
daEmployee = new Project_K4И_01.DataSetEmployeeTableAdapters.
EmployeeTableAdapter();
DataSetEmployeeTableAdapters.JobTitleTableAdapter daJobTitle =
new Project_K4И_01.DataSetEmployeeTableAdapters.
JobTitleTableAdapter();
```

Создайте метод для заполнения объекта dsEmployee из базы данных, после того, как созданы объекты адаптеров данных daEmployee и daJobTitle, а также объект класса DataSetEmployee - dsEmployee (в рассматриваемом примере база данных Employee, созданная в СУБД MS SQL Server 2005). Для заполнения данными dsEmployee из базы данных Employee создадим метод EmployeeFill():

```
public void EmployeeFill()
{
    daJobTitle.Fill(dsEmployee.JobTitle);
    daEmployee.Fill(dsEmployee.Employee);
    MessageBox.Show("Метод Fill отработал");
}
```

В методе `EmployeeFill()` для объектов класса `DataAdapter` применяется метод `Fill`, который производит заполнение таблиц (*JobTitle* и *Employee*) объекта `dsEmployee` данными из базы данных. Метод `Fill` адаптера данных `DataAdapter` требует указания в качестве параметров задания соответствующей таблицы *DataSet*, то есть `dsEmployee.JobTitle` и `dsEmployee.Employee`.

Метод `MessageBox.Show` введен в метод `EmployeeFill` для первоначального тестирования, после которого его нужно убрать.

Вызов метода `EmployeeFill` необходимо добавить в обработчик события `Load` для формы `FormEmployee`, возникающего при нажатии на пункт меню "Сотрудник".

Пример выполнения задания:

1. Свяжите элемент контроля `listBoxEmployee`, в котором должен отображаться список фамилий сотрудников, со столбцом `EmployeeSurname` таблицы `Employee`. Это можно сделать, добавив следующие строки кода в метод загрузки формы `FormEmployee_Load`.

```
this.listBoxEmployee.DataSource = this.dsEmployee1;  
this.listBoxEmployee.DisplayMember = "Employee.EmployeeSurname";
```

В разрабатываемом приложении на *Windows*-форме *FormEmployee* имеются четыре текстовых поля: `textBoxSurname`, `textBoxName`, `textBoxPatronymic` и `textBoxNetName`. Эти текстовые поля предназначены для отображения информации из одной записи таблицы `Employee` набора данных `dsEmployee`. Для того чтобы содержимое текстовых полей автоматически обновлялось при смене записи, их необходимо связать с соответствующими колонками набора данных `dsEmployee`. Связывание можно осуществить, используя свойство `DataBindings` элемента управления `TextBox`.

2. Создайте связь для элемента управления `textBoxSurname` с источником данных, добавив следующую строку кода в метод загрузки формы `FormEmployee_Load`.

```
textBoxSurname.DataBindings.Add("Text", dsEmployee, "Employee.EmployeeSurname");
```

3. Протестируйте добавленный в программу код.

4. Аналогично свяжите текстовые поля `textBoxName`, `textBoxPatronymic` и `textBoxNetName` с источником данных.

В таблице `Employee` значения для атрибута `Access` (доступ) задается в виде символьной строки, значение которой выбирается из списка элемента управления `comboBoxAccess`.

5. Задайте коллекцию выпадающего списка элемента управления `comboBoxAccess` можно задать следующей строкой кода:

```
this.comboBoxAccess.Items.AddRange(new object[] { "не задан", "администратор", "начальник смены", "старший оператор", "оператор", "аналитик" });
```

6. Для заданной записи источника данных (таблица `Employee`) значение столбца `Access` отобразите в элементе контроля `comboBoxAccess`. Сделайте это аналогично тому, как это делалось для элементов управления `TextBox`, задавая свойство `DataBindings`

7. Сформируйте коллекцию выпадающего списка: не задан, активен, выходной, в отпуске, болеет, не работает, помечен как удаленный. Для элемента управления `comboBoxStatus`

В таблице `Employee` значения для атрибута `EmployeeStatus` (статус) задается в виде целого числа (0, 1, 2, 3, 4, 6), однако статус сотрудника должен отображаться в элементе управления `comboBoxStatus` в виде строковых значений в соответствии со значениями его коллекции. В программе необходимо реализовать отображение целочисленных данных из `DataSet` в текстовые значения в элементе контроля `comboBoxStatus`. Для этого необходимо отслеживать изменение позиции в табл. источника данных `dsEmployee` и в соответствии со значением столбца `EmployeeStatus` активизировать требуемый элемент (`Item`) списка `comboBoxStatus`.

8. Объявите объект `bmEmployee` класса `BindingManagerBase` в форме *ormEmployee*:

```
BindingManagerBase bmEmployee;
```

9. Создайте объект `bmEmployee` в конструкторе класса `FormEmployee`, применяя индекатор контента `BindingContext` включив в него связывание с таблицей `Employee`, и добавьте делегат для события, которое формируется при изменении позиции в данной таблице:

```
public FormEmployee( )
```

```
{ InitializeComponent();
```

```
  bmEmployee = this.BindingContext[dsEmployee1, "Employee"];
```

```
  // Добавляем делегата PositionChanged для события - изменение
```

```
  //позиции в табл. Employee DataSet dsEmployee
```

```
  bmEmployee.PositionChanged = new EventHandler(BindingManager-  
Base_PositionChanged); }
```

10. Создайте обработчик для сформированного события, который на основе выбранной строки (`pos`) таблицы `Employee` будет задавать свойству `Text` списка `comboBoxStatus` значение из коллекции `Items` по индексу (`sel`), полученному из столбца `EmployeeStatus` `Employee`.

```
private void BindingManagerBase_PositionChanged(object sender,  
EventArgs e)
```

```
{
```

```
  int pos = ((BindingManagerBase)sender).Position;
```

```
  int sel = (int)dsEmployee.Employee[pos].EmployeeStatus;
```

```
  this.comboBoxStatus.Text = this.comboBoxSta-
```

```
  tus.Items[sel].ToString();
```

```
}
```

11. **Протестируйте добавленный в программу код.**

12. Для задания в элементе контроля `comboBoxJobRole` должности сотрудника необходимо получить данные из родительской таблицы `JobTitle`, с которой таблица `Employee` связана внешним ключом `JobRoleID`. Фактически необходимо осуществить вывод данных из справочника (таблица `JobTitle`) по данным в основной табл. `Employee`.

Свойство `DataBindings` объекта `ComboBox` предоставляет доступ к коллекции `ControlBindingsCollection`. Метод `Add` этой коллекции добавляет в неё привязку.

Перегруженный вариант метода `Add` принимает три аргумента:

`PropertyName` - имя свойства элемента управления, к которому осуществляется привязка;

`DataSource` - имя привязываемого источника данных;

`DataMember` - имя свойства привязываемого источника данных.

13. Свяжите элемент контроля `comboBoxJobRole` с набором данных `JobTitle` (родительская таблица - справочник), в соответствии с кодом, приведенным ниже.

```
comboBoxJobRole.DataSource = this.dsEmployee.JobTitle;
```

```
comboBoxJobRole.DisplayMember = "JobRoleName";
```

```
comboBoxJobRole.ValueMember = "JobRoleID";
```

14. Свяжите `comboBoxJobRole` с полем `JobRoleID` основной таблицы `Employee` (дочерняя таблица), в соответствии со следующим кодом

```
comboBoxJobRole.DataBindings.Add("SelectedValue", dsEmployee, "Employee.JobRoleID");
```

После компиляции и запуска приложения экранная форма будет иметь вид, приведенный на рис.44.

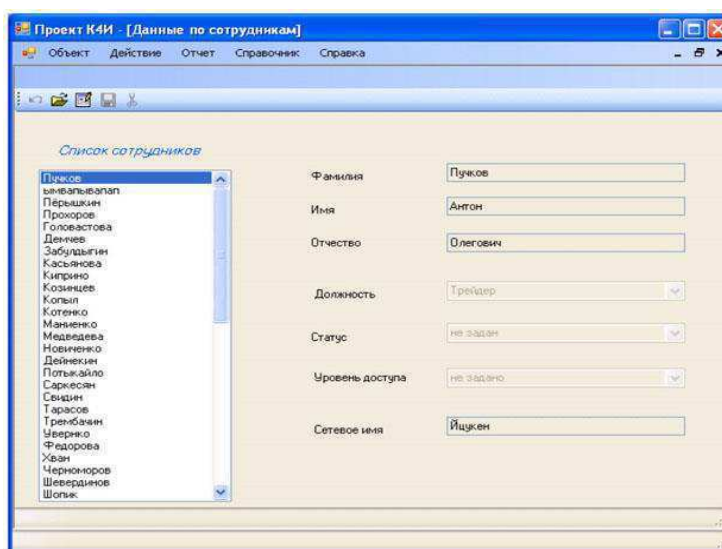


Рисунок 44 – Экранная форма в режиме просмотра (только для чтения)

3. Задание к лабораторной работе

1. Опишите процесс создание класса DataSetEmployee, объектов dsEmployee, daJobTitle и daEmployee, метода Fill.

2. Опишите создания базы данных, включающей две таблицы и соединение проекта с базой данных, формирование новых записей в приложении.

3. Создайте метод для заполнения объекта dsEmployee из базы данных.

4. Опишите процесс связывания элемента контроля listBoxEmployee, в котором должен отображаться список фамилий сотрудников, со столбцом EmployeeSurname таблицы Employee.

5. Выводы по проделанной работе

Контрольные вопросы

1. Как подготовить таблицы базы данных?
2. Как использовать базы данных в приложении?
3. Как создать соединение проекта с базой данных?
4. Как использовать класс DataSetEmployee?
5. Что такое объекты dsEmployee, daJobTitle и daEmployee?
6. Какие функции выполняет метод Fill?
7. Как создать метод для заполнения объекта dsEmployee из базы данных?
8. Какой порядок процесса связывания элемента контроля listBoxEmployee, в котором должен отображаться список фамилий сотрудников, со столбцом EmployeeSurname таблицы Employee?
9. Что представляет собой функция Data.Set?
10. Для выполнения, каких задач предназначены все типы ADO.NET?

Список рекомендуемых источников

1. Проектирование информационных систем. Проектный практикум : учебное пособие / А. В. Платёнкин, И. П. Рак, А. В. Терехов, В. Н. Чернышов ; Тамбовский государственный технический университет. - Тамбов : Тамбовский государственный технический университет (ТГТУ), 2015. - 81 с. : ил., схем. - URL: <http://biblioclub.ru/index.php?page=book&id=444966> (дата обращения 27.02.2020) . - Режим доступа: по подписке. - Библиогр. в кн. - ISBN 978-5-8265-1409-2. - Текст : электронный.
2. Антонов, В. Ф. Методы и средства проектирования информационных систем : учебное пособие / В. Ф. Антонов, А. А. Москвитин ; Северо-Кавказский федеральный университет. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2016. – 342 с. - URL: <http://biblioclub.ru/index.php?page=book&id=458663>. - Режим доступа: по подписке. - Текст : электронный.
3. Влацкая, И. В. Проектирование и реализация прикладного программного обеспечения [Электронный ресурс] : учебное пособие / И. В. Влацкая ; Н. А. Заельская ; Н. С. Надточий. - Оренбург : ОГУ, 2015. - 119 с. – Режим доступа: biblioclub.ru
4. Золотов, С. Ю. Проектирование информационных систем [Электронный ресурс] : учебное пособие / С. Ю. Золотов. - Томск : Эль Контент, 2013. - 88 с. // Режим доступа – <http://biblioclub.ru/index.php?page=book&id=208706>
5. Абрамов, Г. В. Проектирование информационных систем [Электронный ресурс] : учебное пособие / Г. В. Абрамов, И. Медведкова, Л. Коробова. - Воронеж : Воронежский государственный университет инженерных технологий, 2012. - 172 с. //Режим доступа –<http://biblioclub.ru/index.php?page=book&id=141626>
6. Стасышин, В. М. Проектирование информационных систем и баз данных [Электронный ресурс] : учебное пособие / В. М. Стасышин. - Новосибирск : НГТУ, 2012. - 100 с. // Режим доступа – <http://biblioclub.ru/index.php?page=book&id=228774>

8. Романов, В. П. Проектирование экономических информационных систем. Методология и современные технологии [Текст] : учебное пособие / В. П. Романов, Н. З. Емельянова, Т. Л. Партыка ; Российская экономическая академия им. Г. В. Плеханова. - М. : Экзамен, 2005. - 256 с. - (Учебник Плехановской академии). - ISBN 5-472-00742-9.

9. Вендров, А. М. Проектирование программного обеспечения экономических информационных систем [Текст] : учебник для студ. вуз. / А. М. Вендров. - М.: Финансы и статистика, 2000. - 352 с. : ил. - ISBN 5-279-02144-X.

10. Вендров, А. М. Практикум по проектированию программного обеспечения экономических информационных систем [Текст] : учебное пособие / А. М. Вендров. - М. : Финансы и статистика, 2002. - 192 с. - ISBN 5-279-02440-6.

11. Торрес, Р. Дж. Практическое руководство по проектированию и разработке пользовательского интерфейса [Текст] / Р. Дж. Торрес. - М. : Вильямс, 2002. - 400 с.

12. Лапина, Т. И. Информационные системы. Проектный практикум к выполнению и защите ВКР бакалавра по направлению 09.03.02 Информационные системы, 09.03.03 Прикладная информатика/ Т. И. Лапина//Юго-Западный гос. ун-т, ЗАО «Университетская книга»—Курск, 2016.—99с.