**Проектно-конструкторская часть**

Материалы этого подраздела содержат результаты работ, вы­полняемых на этапах эскизного, технического и рабочего проекти­рования, что соответствует проектированию как системы в целом, так и её основных компонентов, определяющих работоспособность конкретного приложения.

Результаты проектирования могут быть расположены в расчет­но-пояснительной записке в следующих подразделах:

1. Разработка структуры приложения.
2. Разработка алгоритмов обработки информации.
3. Логическая схема базы данных
4. Разработка интерфейса взаимодействия пользователя с системой.
5. Разработка архитектуры приложения.
6. Реализация функционирующего приложения.

Материалы проектной части отражаются в графиче­ской части курсовой работы, оформленные в виде чертежей и схем. Все принимаемые технические решения должны быть обоснованы.

Количество и названия подразделов определяются студентом самостоятельно, с целью максимально эффективного освещения процесса проектирования приложения.

Ниже приведены допустимые темы, которые могут быть отражены в каждом из подразделов данной части курсовой работы.

Разработка структуры приложения. Результатом внешнего проектирования является содержательный облик приложения, ко­торый оформлен в виде описания на естественном языке. На осно­ве этой неформальной модели разрабатывается структура приложения (струк­турная схема системы). Она позволяет:

* выявить источники информации и её потребителей. В частности, одним из ключевых источников информации является содержимое базы данных;
* определить состав компонентов системы, среду и платформу для разработки;
* наметить последовательность разработки системы и её ком­понентов.

*Разработка алгоритмов обработки информации.* При необходимости нетривиальных манипуляций с информацией из входных источников следует:

* разработать алгоритмы решения функциональных задач;
* провести тестирование алгоритмов на точность, непротиво­речивость, конечность и другие критерии качества.

В процессе проектирования алгоритмов определяются требуе­мые библиотеки и функции, время и точность решения предлагаемых функциональных задач.

Разработанные алгоритмы должны быть представлены в графи­ческой части работы в виде алгоритмической схемы.

*Логическая схема базы данных.* Логическая схема базы данных имеет смысл в рамках выбранной формальной модели данных (скорее всего это будет реляционная модель, но допустимы целесообразные альтернативы, выбранные в результате исследования). Схема не привязана к конкретной СУБД и физическому размещению. Она обеспечивает инвариантное представление структур данных. В логической модели должны быть описаны все сущности, их атрибуты (допустимо укрупненное представление при большом объеме) и связи.

Необходимо при проектировании схемы данных учитывать жизненный цикл приложения, в частности, обеспечение целостности базы данных по истечении времени, влияние процедур удаления или архивации данных на целостность логической структуры.

Следует обратить внимание на то, что логическая схема не ограничивается ER-моделью. При проектировании логической схемы следует уделить особое внимание описанию условий целостности модели в аспектах: целостность сущностей, целостность отношений, целостность атрибутов, семантическая целостность (нетривиальные ограничения, которые впоследствии должны быть реализованы посредством СУБД (триггеры, встроенные функции) или бизнес-слоем приложения (транзакции). На основании исследования процесса обработки информации из базы данных и используемых алгоритмов должно быть принято решение об оптимальной нормализации логической схемы данных.

*Разработка интерфейса взаимодействия пользователя с системой.* На основании модели пользовательских представлений данных и процессов предметной области должны быть разработаны интерфейсы взаимодействия пользователя с системой, удовлетворяющие современным требованиям по эргономике. Следует обратить особое внимание на то, что в интерфейсе должны быть представлены все данные из пользовательского представления (и только они), а также все допустимые сценарии использования системы.

*Разработка архитектуры приложения.* В подразделе помеща­ются материалы по выбору состава и связей общесистемного программного обеспечения и разрабатываемого приложения. Для обоснованного выбора предпочтительного варианта технического решения следует:

* определить варианты воплощения программ­ного продукта;
* выбрать адекватную платформу разработки и используемую СУБД;
* выбрать методы и средства реализации передачи информации между компонентами системы;
* оценить предполагаемое качество функционирования систе­мы или её компонентов по результатам моделирования на моделях различного уровня;
* выбрать наилучший вариант реализации.

Возможен выбор любой клиент-серверной архитектуры. Допустимо основываться на материале пройденных курсов, но также допустимо использовать новые для студента технологии. Существенно, чтобы изучение новой технологии не осложнило выполнение графика подготовки работы.

*Реализация функционирующего приложения.* Ключевым моментом реализации является построение физической схемы базы данных, основанной на выбранной СУБД, логической модели данных и требованиях к архитектуре и быстродействию системы. Нет необходимости в графическом представлении полной физической схемы. Нет необходимости во включении большого объема предложений DDL в основной текст работы (при необходимости можно включить его в приложение). Вместе с тем при построении физической схемы имеет смысл выделить ключевые запросы к базе данных, обосновать применяемые для их оптимизации методы, описать пользовательские представления данных, встроенные процедуры, транзакции и триггеры (если они имеются).

Исходя из требований предметной области следует проработать и реализовать дисциплину управления контролем доступа, там, где он необходим.

Приложение должно быть законченным, то есть, не допускать ошибок времени исполнения, проходить все обозначенные в интерфейсе сценарии. Именно требование законченности приложения должно определять адекватный функционал на этапе проектирования системы.