

Жизненный цикл информационных систем

Информационная система

— взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели»

Жизненный цикл ИС



— это период времени, который начинается с момента принятия решения о необходимости создания информационной системы и заканчивается в момент ее полного изъятия из эксплуатации.

Полный жизненный цикл информационной системы включает в себя:

- ▶ стратегическое планирование
- ▶ анализ
- ▶ проектирование
- ▶ реализацию
- ▶ внедрение
- ▶ эксплуатацию




Модель жизненного цикла

- некоторая структура, определяющая последовательность осуществления процессов, действий и задач, выполняемых на протяжении жизненного цикла информационной системы, а также взаимосвязи между этими процессами, действиями и задачами.

Каскадная модель жизненного цикла ИС

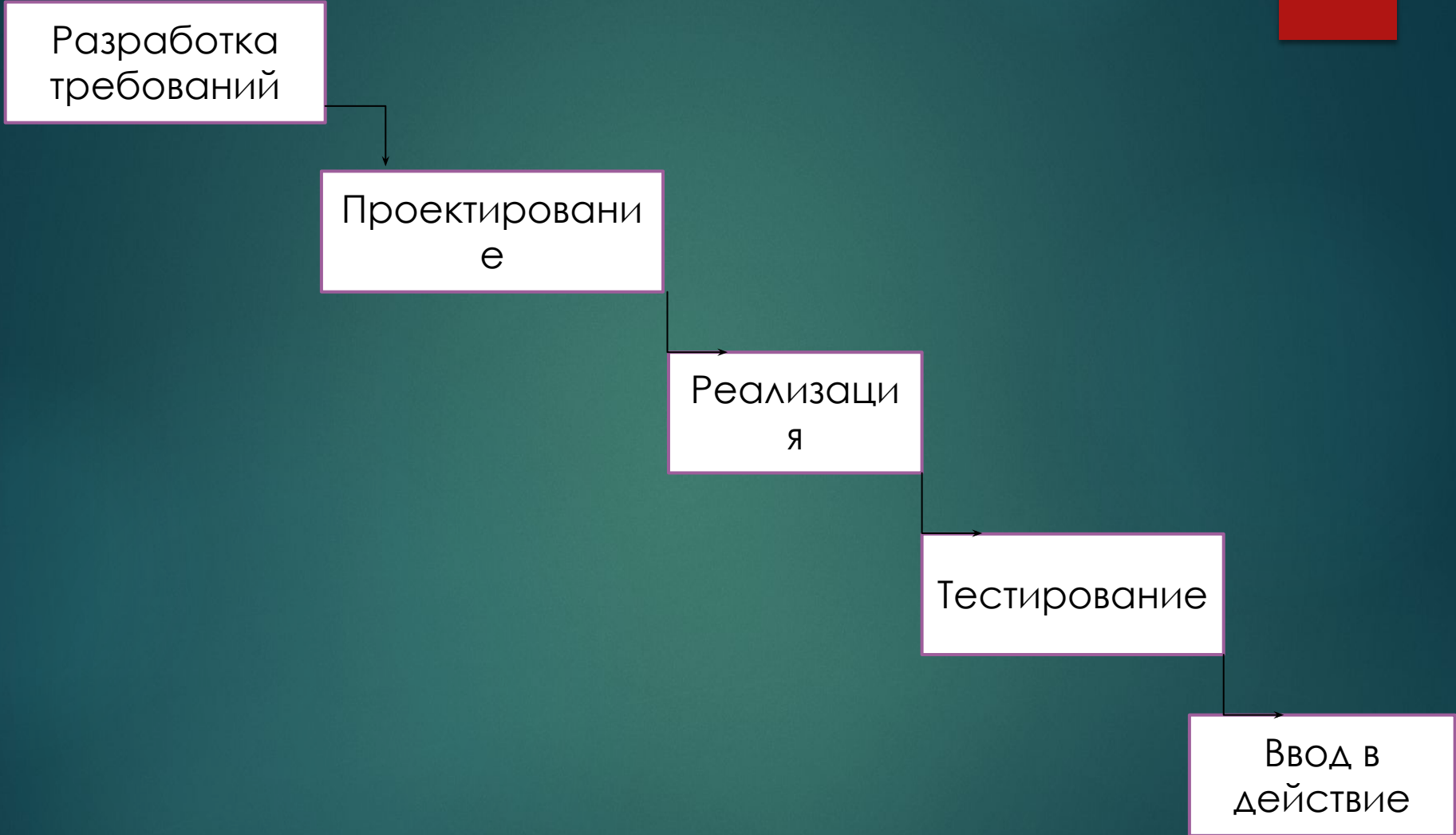
Демонстрирует классический подход к разработке информационных систем в любых прикладных областях.

Предусматривает последовательную организацию работ.



Основной особенностью каскадной модели является разбиение всей разработки на этапы, переход с одного этапа на следующий происходит только после того, как полностью завершены все работы на предыдущем уровне.

Каждый этап завершается выпуском полного комплекта документации, достаточной для того, чтобы разработка могла быть продолжена другой командой разработчиков.



Достоинства каскадной модели:

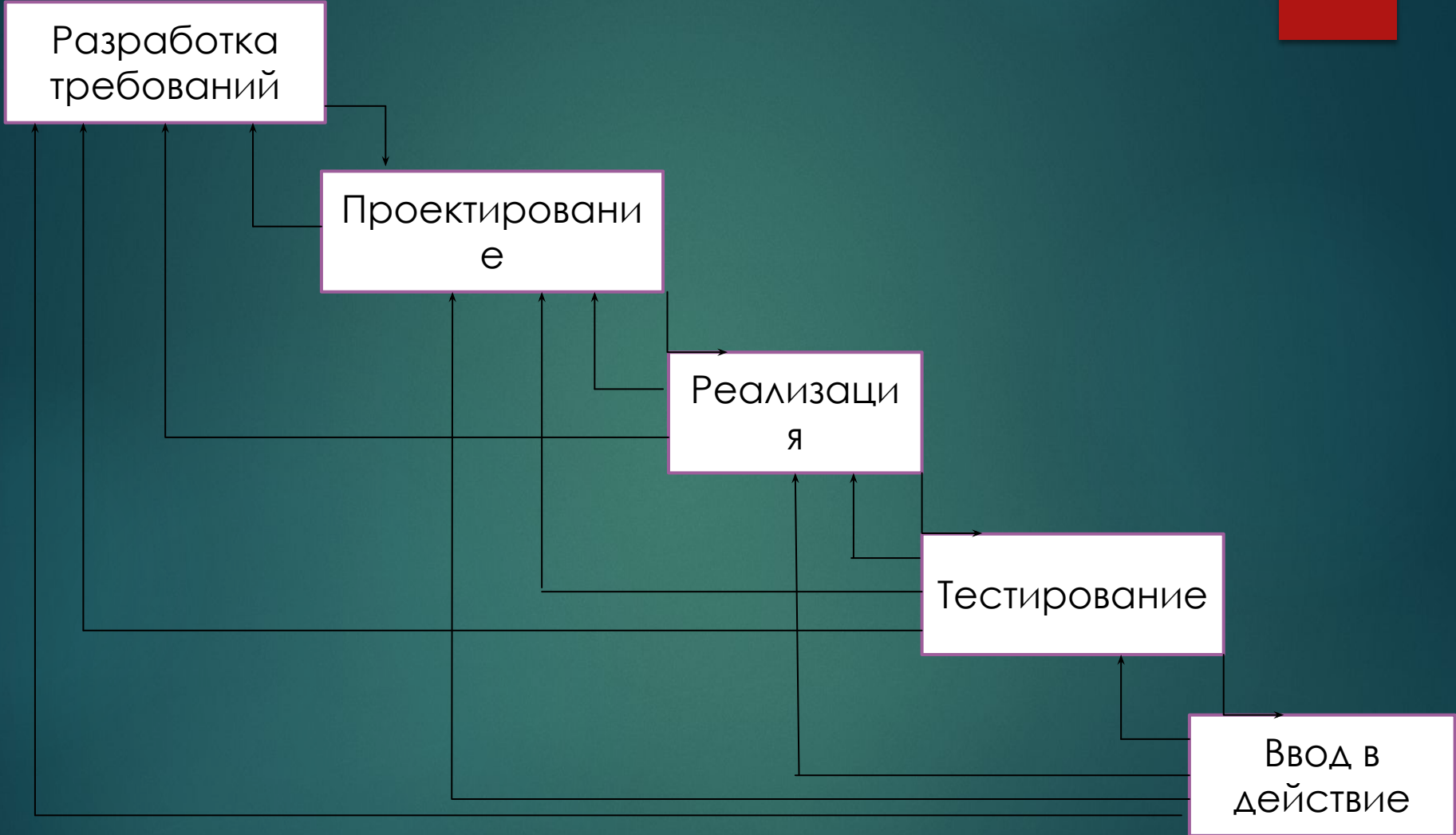
- ▶ на каждом этапе формируется законченный набор проектной документации, отвечающий критериям полноты и согласованности.
- ▶ выполняемые в логической последовательности этапы работ позволяют планировать сроки завершения и соответствующие затраты.

Недостатки каскадной модели:

- ▶ существенная задержка в получении результатов
- ▶ ошибки и недоработки на любом из этапов проявляются на последующих этапах работ, что приводит к необходимости возврата назад
- ▶ сложность параллельного ведения работ по проекту
- ▶ чрезмерная информационная перенасыщенность каждого из этапов
- ▶ сложность управления проектом
- ▶ высокий уровень риска и надежность инвестиций

Поэтапная (Итерационная) модель


Включает в себя промежуточный контроль на любом этапе и межэтапные корректировки. Обеспечивает меньшую трудоемкость по сравнению с каскадной моделью, но время жизни каждого этапа становится равным всему жизненному циклу. Межэтапные корректировки позволяют уменьшить трудоемкость процесса разработки по сравнению с каскадной моделью.



Спиральная модель жизненного цикла

Была предложена для преодоления перечисленных недостатков каскадной модели.

Спиральная модель отличается от каскадной тем, что предполагает итерационный процесс разработки информационной системы. При этом возрастает значение начальных этапов жизненного цикла.



Использование спиральной модели позволяет перейти на следующий этап выполнения проекта, не дожидаясь полного завершения текущего – недоделанную работу можно будет выполнить на следующей итерации.

Главная задача каждой итерации – как можно быстрее создавать работоспособный продукт, который можно показать пользователям системы.

Таким образом, каждый виток спирали соответствует созданию фрагмента или версии программного изделия, на нем уточняются цели и характеристики проекта, определяется его качество, планируются работы на следующем витке спирали.

Спиральная модель ЖЦ



Достоинства спиральной модели

- ▶ Упрощает внесение изменений в проект при изменении требований заказчика
- ▶ Отдельные элементы информационной системы интегрируются в единое целое постепенно
- ▶ Уменьшение уровня рисков
- ▶ Обеспечивает большую гибкость в управлении проектом
- ▶ Упрощает повторное использование компонентов
- ▶ Позволяет получить надежную и устойчивую систему
- ▶ Дает возможность совершенствовать процесс разработки

Недостатки спиральной модели

- ▶ Определение момента перехода на следующий этап

Стадии жизненного цикла ИС

Границы каждой стадии определены некоторыми моментами времени, в которые необходимо принимать определенные критические решения и достигать определенных ключевых целей.

1. Начальная стадия

Устанавливается область применения системы и определяются граничные условия. Для этого необходимо идентифицировать все внешние объекты, с которыми должна взаимодействовать разрабатываемая система, и определить характер этого взаимодействия на высоком уровне. На начальной стадии идентифицируются все функциональные возможности системы и производится описание наиболее существенных из них.

2. Стадия уточнения

На стадии уточнения проводится анализ прикладной области, разрабатывается архитектурная основа информационной системы.

При принятии любых решений необходимо принимать во внимание разрабатываемую систему в целом. Это означает, что необходимо описать большинство функциональных возможностей системы и учесть взаимосвязи между отдельными ее составляющими.

В конце стадии уточнения проводится анализ архитектурных решений и способов устранения главных факторов риска в проекте.

3. Стадия конструирования

На стадии конструирования разрабатывается законченное изделие, готовое к передаче пользователю.

Проводится тестирование и доводка программного комплекса.

По окончании этой стадии определяется работоспособность разработанного программного обеспечения.

Стадия передачи в ЭКСПЛУАТАЦИЮ

Разработанное программное обеспечение передается пользователям. При эксплуатации разработанной системы в реальных условиях часто возникают различного рода проблемы, которые требуют дополнительных работ по внесению корректив в разработанный продукт. Это, как правило, связано с обнаружением ошибок и недоработок.

В конце стадии необходимо определить, достигнуты цели разработки или нет.