

Содержание

1	Информационные ресурсы, идентификационная информация, идентификационные данные, идентификационный атрибут, идентификация	2
1.1	Дополнительная (желательная к прочтению и повторению информация)	2
1.2	Сопутствующая информация	2
2	Аунтефикационная информация, аунтефикация. Многофакторная аунтефикация. Типы аунтефикационных данных	3
2.1	Дополнительная (желательная к прочтению и повторению информация)	3
2.2	Сопутствующая информация	3
3	Авторизация. Определение, практическое значение и применение	3
4	Классические модели процесса: водопадная модель, спиральная модель. Фазы и виды деятельности.	4
4.1	Водопадная модель	4
4.2	Спиральная модель	5

1 Информационные ресурсы, идентификационная информация, идентификационные данные, идентификационный атрибут, идентификация

Информационные ресурсы — это отдельные документы и отдельные массивы документов, документы и массивы документов в информационных системах (библиотеках, архивах, фондах, банках данных, других информационных системах).

Идентификационная информация — это совокупность значений идентификационных атрибутов, которая связана с конкретным субъектом доступа или конкретным объектом доступа.

Идентификационные данные — это совокупность идентификационных атрибутов и их значений, которая связана с конкретным субъектом доступа или конкретным объектом доступа.

Идентификационный атрибут — это атрибут, который характеризует субъект доступа или объект доступа и может быть использован для его распознавания.

Идентификация — это действия по присвоению субъектам и объектам доступа идентификаторов и (или) по сравнению предъявляемого идентификатора с перечнем присвоенных идентификаторов.

1.1 Дополнительная (желательная к прочтению и повторению информация)

Идентификация, аутентификация, авторизация

1.2 Сопутствующая информация

Субъект доступа — это одна из сторон информационного взаимодействия, которая инициирует получение и получает доступ.

Субъектами доступа могут являться как физические лица (пользователи), так и ресурсы стороны информационного взаимодействия, а также вычислительные процессы, инициирующие получение и получающие доступ от их имени.

Объект доступа — это одна из сторон информационного взаимодействия, предоставляющая доступ.

2 Аунтефикационная информация, аунтефикация. Многофакторная аунтефикация. Типы аунтефикационных данных

Аунтефикационная информация — это информация, используемая при аунтефикации субъекта доступа или объекта доступа.

Аунтефикация — это действия по проверке подлинности субъекта доступа и (или) объекта доступа, а так же по проверке принадлежности субъекту и (или) объекту доступа предъявленного идентификатора доступа или аунтефикационной информации.

Многофакторная аунтефикация — это аунтефикация, при выполнении которой используется не менее двух различных факторов аунтефикации.

Типы аунтификационных данных (пока что информация не найдена).

2.1 Дополнительная (желательная к прочтению и повторению информация)

Идентификация, аунтификация, авторизация

2.2 Сопутствующая информация

Субъект доступа — это одна из сторон информационного взаимодействия, которая иницирует получение и получает доступ.

Субъектами доступа могут являться как физические лица (пользователи), так и ресурсы стороны информационного взаимодействия, а также вычислительные процессы, иницирующие получение и получающие доступ от их имени.

Объект доступа — это одна из сторон информационного взаимодействия, предоставляющая доступ.

3 Авторизация. Определение, практическое значение и применение

Авторизация — это предоставление определённому лицу или группе лиц прав на выполнение определённых действий.

4 Классические модели процесса: водопадная модель, спиральная модель. Фазы и виды деятельности.

4.1 Водопадная модель

Водопадная модель является одной из самых старых методологий разработки. Данная методология зиждется на следующих принципах:

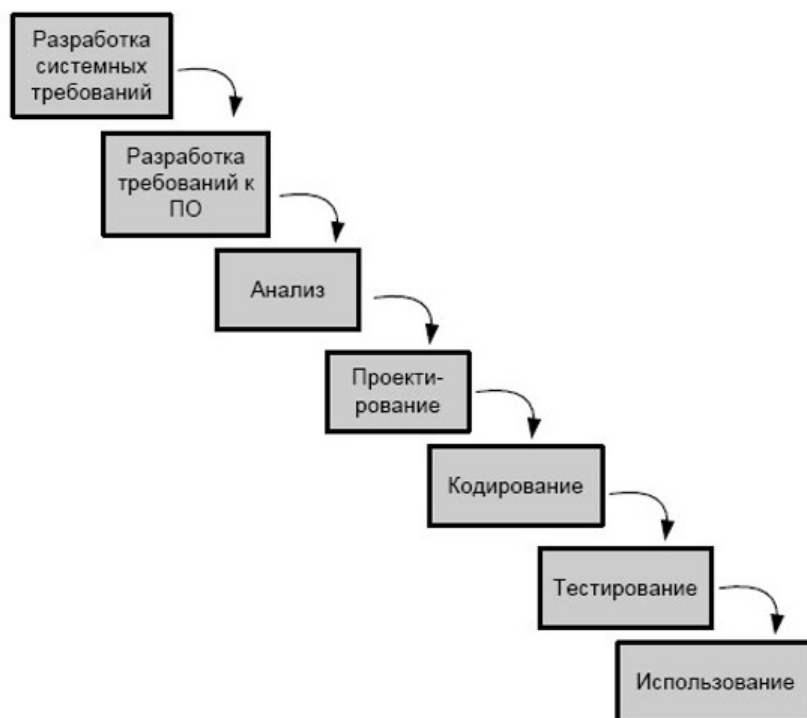
- Полное планирование всех фаз разработки и жёсткое следование ему. В водопадной модели всё планируется и прописывается заранее, каждый этап описан и спланирован, никакие изменения не вносятся до завершения проекта;
- Последовательное выполнение фаз без возможности вернуться на предыдущую фазу;
- Не предполагается, что заказчик имеет доступ к проекту до его сдачи, также предполагается, что правки вноситься не будут.

Плюсы водопадной модели:

- Чёткость и предсказуемость. Всё от требований заказчика, до способов разработки и сроков определено заранее, все следуют плану;
- Понятно, что и как делать. Есть полное ТЗ, которому следует команда, кроме того каждый этап и требования к нему прописаны;
- Подходит крупным проектам, где важно контролировать бюджет. Перед разработкой стоимость и сроки проекта уже известны;
- Подходит для типовых проектов, знакомых команде;
- Заказчик не участвует в каждом этапе разработки. Удобно для тех заказчиков, кто знает чего хочет и у него нет времени для связи с проектом.

Минусы водопадной модели: на мой взгляд все минусы водопадной модели заключаются в невозможности заранее всё спланировать для непредсказуемых проектов или заказчиков. Водопадная модель работает только в тех случаях, когда есть возможность с высокой долей вероятности предусмотреть все варианты развития проекта и спланировать их.

Фазы водопадной модели:



Применяется водопадная модель в гос сфере, так как там важно всё заранее спланировать, в типовых проектах, когда непредвиденных ситуаций благодаря опыту команды не может случиться и в очень крупных проектах тоже, как правило используются данная методология разработки.

4.2 Спиральная модель

Спиральная модель — одна из наиболее важных моделей жизненного цикла разработки ПО, которая обеспечивает поддержку управления рисками. В схематическом представлении она выглядит как спираль с множеством петель. Точное количество витков спирали не известно и может изменяться от проекта к проекту. Каждый цикл спирали называется фазой. Точное количество этапов, необходимых для разработки продукта, может варьироваться менеджером проекта в зависимости от рисков проекта.

Радиус спирали в любой точке представляет собой затраты (стоимость проекта) на данный момент, а угловой размер представляет прогресс, достигнутый на текущий момент.

Spiral Model Спиральная модель и архитектура разработки программного обеспечения



Spiral Model Спиральная модель и архитектура разработки программного обеспечения

Фазы спиральной модели:

Фазы спиральной модели	Действия, выполненные на этапе
Определение целей и определение альтернативных решений	требования собираются от клиентов, а цели определяются, разрабатываются и анализируются в начале каждого этапа. Затем в этом квадранте предлагаются альтернативные решения, возможные для фазы.
Выявление и устранение рисков	во втором квадранте оцениваются все возможные решения, чтобы выбрать наилучшее из возможных. Затем выявляются риски, связанные с этим решением, и риски устраняются с использованием наилучшей стратегии. В конце этого квадранта создается прототип для наилучшего возможного решения.
Разработка следующей версии продукта	в третьем квадранте выявленные функции разрабатываются и проверяются посредством тестирования. В конце третьего квадранта доступна следующая версия программного обеспечения.
Обзор и планирование следующего этапа	В четвертом квадранте заказчики оценивают уже разработанную версию программного обеспечения. В конце концов, начинается планирование следующего этапа.

Плюсы спиральной модели:

- **Обработка рисков:** проекты с множеством неизвестных рисков, возникающих в процессе разработки, в этом случае спиральная модель является лучшей моделью разработки, которой следует следовать, благодаря анализу рисков и управлению рисками на каждом этапе.
- **Подходит для больших проектов:** рекомендуется использовать спиральную модель в больших и сложных проектах.
- **Гибкость в требованиях:** запросы на изменение в требованиях на более позднем этапе могут быть точно включены с помощью этой модели.
- **Удовлетворенность клиентов:** заказчик может наблюдать за развитием продукта на ранней стадии разработки программного обеспечения и, таким образом, привык к системе, используя ее до завершения работы над продуктом.

Минусы спиральной модели:

- **Сложность:** спиральная модель намного сложнее других моделей SDLC.
- **Дорого:** спиральная модель не подходит для небольших проектов, так как она дорогая.
- **Слишком сильно зависит от анализа рисков:** успешное завершение проекта во многом зависит от анализа рисков. Без очень большого опыта невозможно разработать проект с использованием этой модели.
- **Сложность в управлении временем:** поскольку количество этапов неизвестно в начале проекта, поэтому оценка времени очень сложна.