Эта часть работы выложена в ознакомительных целях. Если вы хотите получить работу полностью, то приобретите ее воспользовавшись формой заказа на странице с готовой работой:
https://stuservis.ru/kontrolnaya-rabota/424096

Тип работы: Контрольная работа

Предмет: Информатика

1 ЗАДАНИЕ 3
Модели жизненного цикла информационной системы 3
2 ЗАДАНИЕ 9
Презентация: Эволюция и основы функционирования компьютерных сетей 9
3 ЗАДАНИЕ 13
Анализ статистических данных о пожарах по Липецкой области 13
СПИСОК ЛИТЕРАТУРЫ 20

Под моделями информационных систем понимают структуру, которая состоит из последовательно выполняемых и взаимосвязанные процессов, действий и задач на протяжении всего цикла их жизни.
Самые известные модели жизненного цикла, это следующие:
1. Каскадная модель. Суть ее заключается в том, что она выполняет все этапы проекта, учитывая фиксированный ряд.
2. Спиральная модель. Суть ее заключается в итерационном подходе, когда должна быть разработана информационная система вместе с обратной связью от этапов.
3. Инкрементная модель. Суть ее заключается в разбиении всего проекта, выполняемого жизненного цикла, учитывая всю последовательность итераций, представляющих собой мини-цикл.
Зависит модель жизненного цикла от того, насколько специфична, масштабна и сложна специфика условий и всего проекта, где функционирует сама система модели.
Состоит модель жизненного цикла информационной системы из следующих моментов:
• Стадия;
• Результат работы на каждом этапе;
• Ключевое событие, которое может быть представлено как точкой завершения работы, так и принятия решения.
Разное состояние системы модель жизненного цикла может отразить прямо с самого начала ее появления, а также по необходимости работы всей информационной системы, а кончается она моментом, когда она полностью выходит из использования ее системой.
Если вернуться к типу моделей жизненного цикла, то можно подробнее рассмотреть каждый из них. Первая модель – каскадная, необходима для последовательного выполнения каждого этапа проекта в порядке, строго фиксированной системой. Когда система переходит на следующий этап, тогда полностью завершается данный тип модели.
Первый этап строится на исследовании проблемы, которую нужно решить, здесь нужно четка сформировать каждое требование клиента. В качестве результата можно представить техническое задание, которое бы устроило каждую из сторон, задействованную в жизни цикла модели.
Во время второго этапа необходимо разработать решения проекта, которые бы удовлетворили каждому требованию, указанному в задании. В качестве результата этапа можно привести проектную документацию, в которой содержатся все требуемые данные, необходимые для того, чтобы проект был реализован.
На третьем этапе модель должна быть реализована. Именно сейчас пришло время для того, чтобы разработать программное обеспечение, например, закодировать информацию, учитывая пункты 1 и 2. Методами реализации проекта не выступает какие-то определенные, можно выбрать любые подходящие. В качестве результата можно получить программный продукт.
Далее идет четвертый этап, когда происходит тестирование продукта, полученного во время третьего этапа. Он должен соответствовать требованиям, которые были заявлены в техническом задании. Если использовать демо-версию, то можно понять, что не работает, а где возникают ошибки, чтобы все исправить на этом этапе. В качестве результата появляется готовая система для эксплуатации.
На последнем этапе сдается готовый проект, начинается его рабочая эксплуатация. На этапе главной задачей является документальное подтверждение, из которого видно, что были учтены требования клиента

полностью. В качестве результата появляется работающая информационная система вместе с сопроводительной документацией, а также все это должен подтвердить заказчик, который принимает систему в работу.
Выделим плюсы от применения каскадного подхода:
• Каждый этап сопровождается законченным набором документов, которые отвечают всей полноте и согласуется с требованиями заказчика;
• Все этапы выполняются логически последовательно, учитывая назначение каждого этапа, получают при этом смету затрат на получении каждого результата;
• Строятся простые информационные системы, позволяющие просчитать с точностью до рубля все затраты и учесть каждое требование заказчика.
Среди минусов можно выделить:
• Большую задержку в момент выдачи результата;
• Тот факт, что если была допущена ошибка на предыдущем этапе, об этом становится известным только на следующих этапах, поэтому приходится возвращаться на шаг, а то и на два назад;
• Практически не получается выполнять параллельную работу по созданию модели;
• На каждом этапе происходит большая перегруженность информацией;
• Проектом управлять в таком типе модели жизненного цикла сложно;
• Инвестирование проекта может оказаться неоправданно дорогим, к тому же высок уровень потери вкладываемых денег.
Теперь поговорим про поэтапную модель, когда используется промежуточный контроль. В этом случает разрабатывается информационная система с помощью отдельных шагов для каждого цикла, получая при этом обратную связь между этапами.
Выходит, что корректировку можно вносить, правильно оценивая то, как влияет на результат предыдущий, другими словами, выявлено взаимовлияние на каждый этап. Выходит, что при поэтапной нагрузке модель живет на каждом этапе вплоть до окончательного результата.
Остановимся на спиральной модели жизненного цикла. Благодаря такой поэтапной модели, у которой промежуточный контроль, можно преодолеть все недостатки в каскадном типе модели.
Во время анализа и проектирования учитывается техническое решение и степень того, как удовлетворены потребности клиента, что проверить можно при создании прототипов. На каждом витке спирали в результате можно получить работоспособный фрагмент, по-другому, очередную систему. Так можно будет знать о требованиях и целях с характеристиками проекта, определив качественную разработку, план для работы очередного витка. Так происходит углубление и последовательно конкретика деталей проекта. В итоге, можно выбрать среди нескольких версий тот продукт, который удовлетворит требованиям клиента полностью.
Выходит, что если использовать спиральную модель, тогда можно переходить на очередной этап, даже если на предыдущем не был еще получен результат. Так недочеты уходят во время выполнения следующего этапа. Можно отметить, что для итераций выступает одна задача – это быстро создать продукт, готовый для эксплуатации, что уже видно клиенту.
Итак, видим, что процесс уточнения и дополнения каждого проекта становится проще. Выделим плюсы спиральной модели:
1. Если заказчик изменит требования, их легко внести даже в почти готовый проект.
2. Если используется спиральная модель, тогда можно отдельный новый элемент интегрировать в одно целое, делая это постепенно. Получается, что начинается работа с меньшим количеством элементов, что существенно уменьшает проблемы в работе.
3. Уменьшается уровень риска. Риск обнаруживается при интеграции, поэтому эти два соседних преимущества помогают избавиться от них сразу. Риски максимальны только в начале работы над проектом, но чем дальше от начала и ближе к завершению создания программы они уменьшаются. Для каждого вида модели данный момент является плюсом, но в спиральной модели скорость уменьшения рисков самая высокая. Выходит, такой эффект из-за того, что итерация на первом этапе позволяет сделать интеграцию уже вначале разработки проекта.
4. Еще одно преимущество заключается в том, итерационная разработка обеспечивает серьезную гибкость во время управления проектом, когда можно изменить любую его модель. Например, можно сократить сроки разработки за счет снижения функциональности системы или использовать в качестве составных частей системы продукцию сторонних фирм вместо собственных разработок.
5. Итерационный подход спиральной модели жизненного цикла упрощает повторное использование

компонентов (реализует компонентный подход к программированию). Это обусловлено тем, что гораздо проще выявить (идентифицировать) общие части проекта, когда они уже частично разработаны, чем пытаться выделить их в самом начале проекта. Анализ проекта после проведения нескольких начальных итераций позволяет выявить общие многократно используемые компоненты, которые на последующих итерациях будут совершенствоваться.
6. Итерационный подход дает возможность совершенствовать процесс разработки – анализ, проводимый в конце каждой итерации, позволяет проводить оценку того, что должно быть изменено в организации разработки, и улучшить ее на следующей итерации.
Среди недостатков спиральной модели можно выделить следующие:
• Основная проблема спирального цикла — определение момента перехода на следующий этап. Для ее решения необходимо ввести временные ограничения на каждый из этапов жизненного цикла. Иначе процесс разработки может превратиться в бесконечное совершенствование уже сделанного.
• При итерационном подходе полезно следовать принципу «лучшее — враг хорошего». Поэтому завершение итерации необходимо проводить строго в соответствии с планом, даже если не вся запланированная работа закончена.
• Планирование работ обычно проводится на основе статистических данных, полученных в предыдущих проектах, и личного опыта разработчиков.

1. Бабешко В.Н., Медведева В.А., Кищенко И.И. Гетерогенные распределенные системы в туманных сетевых инфраструктурах.
2. Бирюкова, Н.Б. Информатизация процесса профессиональной подготовки специалистов в сфере экономической безопасности Российской Федерации / Н.Б. Бирюкова, А.В. Новиков – Москва: Наука, 2016. – 352 с.
3. Буренин С.В., Корочкин М.А. Тесты по информатике с ответами и комментариями. Раздел1. Аппаратная часть вычислительной системы. Компьютерные сети. Интернет: Учебно-метод. Пособие / С.В. Буренин, М.А. Корочкин. – Иваново: ООНИ ИвИ ГПС МЧС России, 2011. – 64 с.
4. Вендров, А.М. Практикум по проектированию программного обеспечения экономических информационных систем / А.М. Вендров. – М.: Финансы и статистика, 2016. – 911 с.
5. Верещагина, Е.А. Корпоративные информационные системы. Учебно-методический комплекс / Е.А. Верещагина. – М.: Проспект, 2015. – 318 с.
6. Гончаров, В.А. Методы оптимизации. Учебное пособие / В.А. Гончаров. – М.: Юрайт, 2015. – 417 с.
7. Гусева, Г.С. Экономическая информатика. Ч. 2. Электронные Таблицы Microsoft Excel И Их Использование Для Экономических Расчетов / Гусева Г.С. – Москва: ИЛ, 2015. – 451 с.
8. Информатика: Базовый курс: учебник для студентов технических специальностей и преподавателей вузов. Изд. 2-е / С.В. Симонович [и др.]; под ред. С.В. Симоновича. – СПб.: Питер, 2009. – 640 с.
9. Коноплева, И.А. Информационные технологии / И.А. Коноплева, О.А. Хохлова, А.В. Денисов. – Москва: ИЛ, 2013. – 328 с.
10. Олифер В. Г. Компьютерные сети: принципы, технологии, протоколы: учеб. пособие для высш. учеб. заведений. – 4-е изд.– СПб.: Питер, 2010. – 944 с.
11. https://ria.ru/20231030/pozhary-1906096387.html?ysclid=lsrxws9z17107137000
12. https://lipetsktime.ru/news/incidents/na_territorii_lipetskoy_oblasti_v_2023_godu_proizoshlo_675_pozharov/?ysclid=lsrxhg51

Эта часть работы выложена в ознакомительных целях. Если вы хотите получить работу полностью, то приобретите ее воспользовавшись формой заказа на странице с готовой работой:
https://stuservis.ru/kontrolnaya-rabota/424096