



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика, искусственный интеллект и системы управления
КАФЕДРА Системы обработки информации и управления

Домашнее задание №1
по курсу «Технологии разработки программного обеспечения»

Под редакцией к.т.н. доц. Виноградовой М.В.

Москва, 2024 г.

1. ЗАДАНИЕ

Выполнить сбор и анализ требований в соответствии с унифицированной моделью разработки RUP.

1.1. Цель работы

- Изучить унифицированный процесс разработки (RUP);
- Приобрести умения построения моделей сбора и анализа требований;
- Получить навыки построения, согласования и преобразования моделей.

1.2. Средства выполнения

- Среда моделирования, например, Sparx Enterprise Architect, StarUml или аналогичные.

1.3. Пункты задания для выполнения

В домашнем задании можно использовать материалы, полученные при выполнении лабораторных работ.

Создать новый проект типа UML.

Добавить в проект модель требований.

В модель требований добавить:

1. Перечень функциональных требований.
2. Перечень нефункциональных требований.
3. Глоссарий понятий.
4. Диаграмму классов предметной области.
5. (дополнительно) Диаграмму бизнес-процессов, перечень и описания бизнес-актеров и бизнес-процессов.
6. (дополнительно) Диаграмму деятельности для иллюстрации работы бизнес-процессов.
7. Описания актеров и прецедентов системы.

8. Начальную диаграмму прецедентов.
9. Спецификацию главного прецедента.
10. (расширенное) Диаграмму деятельности по главному прецеденту (п.9).
11. Экранные формы для главного прецедента (соответствие п.9, 10, 11).
12. (расширенное) Экранные формы для основных прецедентов (3-7 шт.)
13. Уточненную диаграмму прецедентов.

Добавить в проект модель анализа.

В модель анализа добавить:

1. Диаграмму трассировки прецедентов в кооперации.
2. Обзорную диаграмму классов-сущностей.
3. Обзорную диаграмму граничных классов.
4. Обзорную диаграмму управляющих классов.
5. Для трех основных (нетривиальных) коопераций диаграммы классов-участников и диаграммы последовательностей.
6. (расширенное) Обобщенную диаграмму классов анализа с указанием их атрибутов и методов.
7. Диаграмму распределения классов анализа по пакетам.
8. Диаграмму зависимостей пакетов анализа.

1.4. Содержание отчета

- Титульный лист;
- Цель работы;
- Задание;
- Артефакты модели требований;
- Артефакты модели анализа;
- Вывод;
- Список используемой литературы.

1.5. Контрольные вопросы

1. Что такое унифицированный процесс разработки? Для чего он предназначен и каковы его особенности?
2. Почему RUP является итерационно-инкрементным?
3. Что такое итерация и какие рабочие процессы включены в нее?
4. Определите действия каждого из рабочих процессов.
5. Каково назначение рабочего процесса определения требований? Какие артефакты создаются в процессе выполнения рабочего процесса?
6. Что такое требование? В чем заключаются различия функциональных и нефункциональных требований? Приведите примеры.
7. Что такое контекст системы? Из чего состоит его описание?
8. Что содержит и как составляется модель предметной области.
9. Из каких действий состоит определение функциональных требований?
10. Что такое актер и прецедент? Как их идентифицируют?
11. Что такое прототип пользовательского интерфейса? Как его составляют?
12. Объясните содержание модели прецедентов, ее элементы и диаграммы.
13. Что содержится в спецификации прецедента? Для чего ее составляют?
14. Как и зачем выполняют структурирование модели прецедентов? Что означают связи включения и расширения и в чем их различия?
15. Объясните назначение рабочего процесса анализа требований. Какие артефакты создаются в процессе выполнения рабочего процесса?
16. Из чего состоит модель анализа, из каких элементов и диаграмм.
17. Расскажите, как выполняют анализ архитектуры.
18. Что такое пакет и сервисный пакет? Как их идентифицируют?
19. Назовите стереотипы классов анализа.
20. Что такое граничный класс и как его выявить? Каковы его атрибуты?
21. Что такое управляющий класс и как его выявить?
22. Что такое класс сущности и как его выявить? Каковы его атрибуты?
23. Как выполняют анализ коопераций? Что содержат диаграммы классов-участников и диаграммы их взаимодействия?
24. Как выполняют анализ класса? Откуда находят атрибуты и ответственности классов анализа, связи между ними?
25. Как выполняют анализ пакетов? В чем его цель?
26. Объясните, откуда появляются и что означают зависимости пакетов.

1.6. Список источников

1. Унифицированный процесс разработки программного обеспечения: учебное пособие / Виноградова М.В., Белоусова В.И. – М.: МГТУ им.Н.Э. Баумана. – 2015 г. <http://ebooks.bmstu.ru/catalog/193/book1303.html>
2. Орлов С.А., Цилькер Б.Я. Технологии разработки программного обеспечения. СПб.: Питер, 2012. 609 с.
3. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения: пер. с англ. СПб.: Питер, 2002. 493 с.