

UML документация



Ревизия 1.0

📅 Dec 4, 2021

Андреев Вячеслав

TABLE OF CONTENTS

UML диаграммы

Как будет организован процесс документирования

ГД диаграммы(Все обязательны при проработке системы)

Диаграммы программистов

Дополнительно

UML диаграммы

UML - это универсальный язык проектирования систем и бизнес-логики. Служит хорошим инструментом для составления тз и документации проекта.

Ссылка на инфу(смотреть закреп)



<https://www.youtube.com/watch?v=0I9aIP5gKCg&list=LL&index=1>

Статья на хабре.

<https://habr.com/ru/post/458680/>

Скачиваем прогу, в которой будем работать(выбираем Community Edition)

<https://www.softwareideas.net/>

Ссылка на реп.

http://185.180.231.95:3000/Covenant/Tenebrium_UML

Как будет организован процесс документирования

1. Каждый отдел работает со своим набором диаграмм
2. Каждая система является самостоятельным проектом диаграмм
3. Над системой сначала работают ГД, строят свои диаграммы, после чего задача уходит программистам, которые по выполнению строят свои диаграммы в том же проекте диаграмм.
4. При проектировании новой системы создается новая ветка, которая удаляется после завершения этапа работы и мерджа в master
5. Завершенная работа дополнительно конвертируется в pdf и заливается на google-disk(Development/UML/ПАПКА_СИСТЕМЫ) для быстрого доступа и организации ссылок на диаграмму в Диз-доках. Репозиторий служит для хранения исходников.

ГД диаграммы(Все обязательны при проработке системы)

1. Use-Case - проработка взаимодействия механики с пользователем
2. Activity-Diagram - проработка поведения системы на высоком уровне абстракции. Тут не нужно писать как работает, тут нужно писать что делает и какие особенности могут быть.(Поведение всей системы в целом)
3. State Diagram - более подробно про каждый требуемый блок функционала. Высокие абстракции поведения в разных условиях.(Похож на Activity diagram, но тут про каждый блок действия)
4. User interface - для макетов UX/UI

Диаграммы программистов

1. Class diagram - описывает архитектуру системы и связи классов
2. Deployment Diagram - описывает связи с другими компонентами/системами(для каждой системы рисовать свою мини диаграмму связей с другими системами). Существует еще 1 общая, которая расширяется по мере проработки систем.
3. Component Diagram - описание взаимодействия элементов внутри одной системы

4. Composite Structure Diagram - тоже как Deployment Diagram описывает связь систем, но уже на более низком уровне
5. Sequence diagram - описывает логику поведения конкретного объекта. Строится по реализации в коде.
6. Interaction Overview Diagram - композиция взаимодействий Sequence diagram

Дополнительно

State Diagram - для описания state-machine, если используется.

- Обязательны: 1,2
- Когда архитектура выше средней по сложности восприятия: + 3,4
- Для сложных низкоуровневых моментов: + 5, 6
- Помимо UML диаграмм в силе остаются и текстовые доки. В них гд прописывают общий поток мыслей, ссылки на рефы и т.д.

Программисты описывают текстом общее представление о системе, скрины из движка для ГД и пояснительные записки к использованию системы силами ГД.