Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЁТ

к лабораторной работе № 4

|  |  |
| --- | --- |
| Студенты: | Д.М.Дерюгин  А.Ю.Губаревич |
| Проверил: | Е.В. Богдан |

Минск 2024

1 Ведущий-ведомый

Стиль "ведущий-ведомый" в контексте программирования предполагает разделение ролей между участниками проекта, где один (ведущий) принимает решения и направляет процесс, а другой (ведомый) выполняет задачи под его руководством.

В данном случае, Дерюгин (M) выступает ведущим, имея более глубокие знания о программировании и разработке, а Губаревич (S).

Плюсы подхода "ведущий-ведомый":

Распределение ролей: Ясно определены обязанности и ответственность каждого участника, что способствует более эффективному выполнению задач.

Фокус на экспертизе: Каждый участник может сосредоточиться на своей области знаний и навыков, что способствует более качественному результату.

Обучение и развитие: В процессе работы ведомый (S) имеет возможность учиться у ведущего (M), расширяя свои знания и опыт.

Недостатки подхода:

Ограниченная гибкость: Распределение ролей может привести к снижению гибкости в реагировании на изменения требований или проблемы в процессе разработки.

Односторонний взгляд: Ведущий может упустить некоторые аспекты проекта, не учитывая точку зрения ведомого, что может привести к недоразумениям или недочетам.

Зависимость от ведущего: В случае отсутствия ведущего, работа над проектом может замедлиться или приостановиться.

Итоги:

В целом, подход "ведущий-ведомый" может быть эффективным, особенно при работе над проектами с четко выделенными областями ответственности. Однако для достижения оптимальных результатов необходимо учитывать потенциальные недостатки и стараться совершенствовать коммуникацию и взаимодействие между участниками проекта.

2 ПИНГ-ПОНГ

В стиле "пинг-понг" программисты работают в парах, где каждый из них чередует роль пинга (написание кода) и понга (рефакторинг, тестирование, обзор кода).

Преимущества подхода "пинг-понг":

Быстрая обратная связь: Каждый программист имеет возможность мгновенно получать обратную связь от своего партнера, что позволяет быстро выявлять ошибки и улучшать код.

Совместное обучение: Взаимное обучение и обмен знаниями стимулируются в процессе пинг-понга. Каждый участник может учиться у партнера, что способствует повышению качества кода и навыков программирования.

Распределение ответственности: Обязанности между участниками равномерно распределены, что способствует более эффективному использованию ресурсов и уменьшает риск усталости или перегрузки одного участника.

Недостатки подхода:

Необходимость согласования: Для эффективной работы пары программистов необходимо постоянное согласование действий и решений, что может замедлить процесс в случае разногласий или разночтений.

Зависимость от коммуникации: Успех подхода "пинг-понг" во многом зависит от качества коммуникации и взаимодействия между участниками. Недостаточная коммуникация может привести к неэффективной работе и возникновению конфликтов.

Риск однообразия: При длительной работе в паре может возникнуть риск утомления или однообразия в подходах к решению задач, что может отразиться на качестве кода.

Итоги:

Подход "пинг-понг" является эффективным способом совместной работы программистов, способствующим повышению качества кода, обмену знаниями и распределению ответственности. Однако для успешной реализации необходимо уделить внимание коммуникации, согласованию действий и предотвращению рисков, связанных с возможным утомлением или конфликтами.

3 вывод

Вывод: в ходе разработки способом ведущий-ведомый мы реализовали создание класса обработки звука. Нам показалось что ПП это довольно сомнительный способ, но всё-таки разработка шла чуть-чуть быстрее. Стиль пинг-понг показался более приятным. Однако в дальнейшем ПП использовать мы не планируем.