

Отчет по парному программированию

Котики



Ведущий-ведомый:

- a. Виталий Герасимович - ведомый
- b. Андрей Матусевич - ведущий

На равных:

- c. Владислав Ковалев
- d. Сергей Твердохлеб

Пинг-понг:

- 1.
 - a. Виталий Герасимович
 - b. Владислав Ковалев
- 2.
 - a. Сергей Твердохлеб
 - b. Андрей Матусевич

Case #0

Стиль ПП: ведущий-ведомый
Ведомый: Виталий Герасимович
Ведущий: Андрей Матусевич
Что разрабатывали: парсинг Json для forecast'ов.

Виталий

Что понравилось:

Мне понравилось, что я могу сосредоточиться полностью на написании кода и всю теоретическую часть оставить на ведущего. Также могу приносить пользу не имея полных знаний в той или иной области, но в нашем случае это было программирование под Android и работой с Json'ами. Что нашли общий язык и при этом смог узнать новые моменты в процессе данного метода (парного программирования).

Что не понравилось:

Нужно найти подходящее время для обоих участников.

Андрей

Что понравилось:

Что я могу сосредоточиться на том, "что" нужно реализовать, а то "как" это реализовать могу остановить на ведомого. Удалось код сделать понятным не только для себя, но и для другого члена команды. Повысилось качество кода, потому что начали одновременно смотреть на код с двух разных сторон.

Что не понравилось:

Нужно найти подходящее время для обоих участников.

Проблемы

Из-за совместного написания кода, его количество вышло меньше, нежели если мы писали его по отдельности.

Выводы

Эту практику стоит применять, но не на постоянной основе, ибо продуктивность меньше. Но она идеально подходит для повышения уровня знаний, а также уровня понимания разрабатываемого проекта и между участниками

Case #1

Стиль ПП: на равных
Кто: Владислав Ковалёв
Твердохлеб Сергей
Что разрабатывали: отображение давления, влажности, ветра на завтра.

Владислав

Что понравилось:

Экономится время на разбор кода напарника. Хорошая возможность делиться опытом. Улучшение взаимоотношений в команде. Сережа<3

Что не понравилось:

Приходится подстраиваться под стиль напарника. Нужно найти подходящее время для обоих участников.

Сергей

Что понравилось:

Возможность лучше понять код члена своей команды, а также ускоряется время разработки за счёт постоянного обмена опытом. И улучшение взаимоотношений в команде. Владик<3

Что не понравилось:

Приходится подстраиваться под стиль напарника. Нужно найти подходящее время для обоих участников.

Проблемы

Уменьшение производительности. Отличия в стиле программирования.
Неопределенность в том, какую часть работы должен выполнять каждый программист.

Выводы

Данный метод ПП имеет свои достоинства и недостатки. В целом, для людей, хорошо понимающих друг друга и работавших вместе и раньше, это отличная штука.

Case #2

Стиль ПП: пинг-понг
Кто: Владислав Ковалёв
Виталий Герасимович
Что разрабатывали: отображение давления, влажности, ветра на сегодня.

Виталий

Что понравилось:

Мне понравилось, что я могу сосредоточиться полностью на написании теста, а последующую всю работу оставить на “напарника”.

Что не понравилось:

Нужно найти подходящее время для обоих участников.

Владислав

Что понравилось:

Что я могу сосредоточиться на написании кода, не тратя времени на проверку правильности его работы. Что я могу проследить ход мысли “напарника” и посмотреть на всё это с другой стороны и тем самым ускорить свою работу и к тому же мне становится проще понимать членов своей команды.

Что не понравилось:

Нужно найти подходящее время для обоих участников. Было трудно проследить ход мысли “напарника”.

Проблемы

Из-за совместного написания кода, его количество вышло меньше, нежели если мы писали его по отдельности. Бывает трудно проследить ход мысли “напарника”. Иногда возникают споры.

Выводы

Повышается качество кода и его читабельность, позволяет получить большую продуктивность нежели в “ведомом-ведущем” (работают сразу двое) и “на равных” (потому что есть конкретное разделение обязанностей).

Case #3

Стиль ПП: пинг-понг
Кто: Сергей Твердохлеб
Андрей Матусевич
Что разрабатывали: прогноз на неделю

Сергей

Что понравилось:

Экономится время на написание тестов и проверку кода. Меньше шанс не увидеть ошибку.

Что не понравилось:

Нужно найти подходящее время для обоих участников.

Андрей

Что понравилось:

Не было боязни писать тесты, так как знал, что код будет писать другой человек. Т.е. написанные тесты были более безжалостными. А если бы и код, и тесты писал я, то было бы морально трудно написать действительно хорошие тесты, так как ты знаешь, что тебе потом придется писать код, который соответствует этим тестам.

Что не понравилось:

Нужно найти подходящее время для обоих участников.

Проблемы

В результате было написано меньше кода, чем когда два разработчика пишут отдельно. Человеку, который в данный момент не пишет, легко отвлечься.

Выводы

Так как написанием тестов и кода занимались разные люди, то в итоге код получился более надежным, уменьшилось число ошибок. Однако, на выходе кода получилось меньше. Поэтому, данный подход следует использовать при реализации тех частей приложения, где в первоочередном порядке необходимо минимизировать вероятность ошибок.