МИНИСТЕРСТВО науки и высшего ОБРАЗОВАНИЯ РОссИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(национальный исследовательский университет)»

Институт № 3 «Системы управления, информатика и электроэнергетика»

Кафедра № 304 «Программное обеспечение средств вычислительной техники и автоматизированных систем»

**Отчёт по практической работе №5**

**по курсу «Технологии разработки программного обеспечения»**

**по теме: «Рефакторинг кода»**

**Вариант 1**

Выполнил:

Благушин Н.А.

Группа: М3О-107СВ-24

Приняли:

Титов Юрий Павлович

Москва 2024 г.

**Задание**

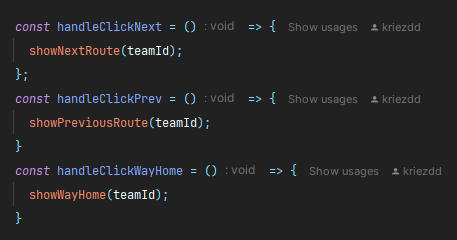
При выполнении практической работы студенты учатся проводить рефакторинг существующего кода, кода программного обеспечения, созданного в результате выполнения творческой работы. Отчетностью по данной практической работе служит изменённый код с указанием «было-стало». Для каждого изменения дается словесное описание причин изменения и достоинств, полученных в результате изменения кода. Также требуется дать словесную (или числовую, приблизительную оценку) технического долга для функции/объекта/модуля в котором произведен рефакторинг. При изучении рефакторнга важно рассмотреть различные варианты изменения кода. Так как программное обеспечение, написанное студентами, обычно не обладает сложными зависимостями, то студенты будут стремиться выполнить в результате рефакторинга только простые методы, по типу «Переименование метода (Rename Method)», то в результате рефакторинга студентам разрешается брать способы только из тех, что приведены в вариантах. При этом не обязательно использовать все способы рефакторинга, а можно многократно повторить один из них.

Вариант1.

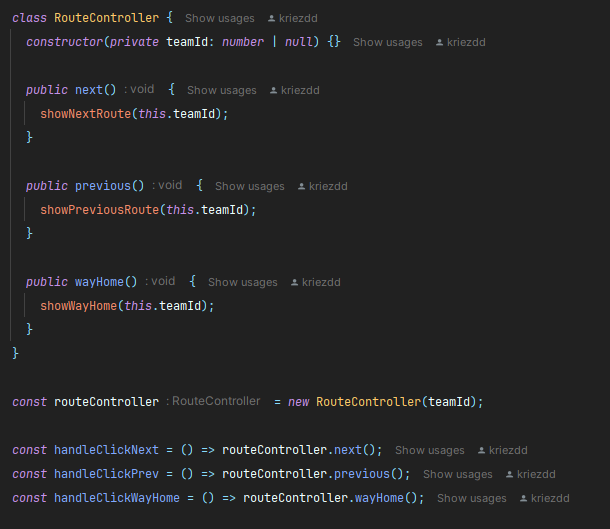
1. Выделение метода (Extract Method)
2. Замена метода объектом методов (Replace Method with Method Object)
3. Удаление посредника (Remove Middle Man)
4. Инкапсуляция поля (Encapsulate Field)

Замена метода объектом методов (Replace Method with Method Object)

Было:



Стало:



Причина изменения:

Вместо трёх отдельных функций handleClickNext, handleClickPrev, handleClickWayHome, был создан объект RouteController, который инкапсулирует методы маршрутов.

Достоинства:

Упрощение вызова маршрутов, что позволяет масштабировать приложение (например, добавлять новые действия с маршрутом).

Инкапсуляция поля (Encapsulate Field)

Было:



Стало:



Причина изменения:

Инкапсулированное управление состоянием teamId с помощью функции useTeam, которая предоставляет интерфейс для установки и сброса ID команды.

Достоинства:

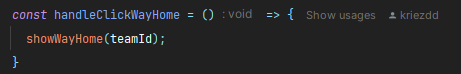
Упрощается управление состоянием.

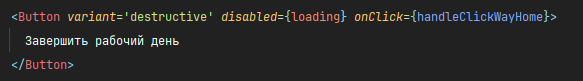
Логика работы с teamId изолирована и понятна.

Удобство при добавлении дополнительных методов работы с teamId (например, валидации или асинхронных обновлений).

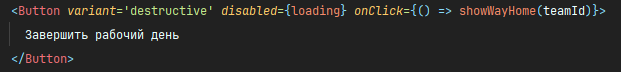
Удаление посредника (Remove Middle Man)

Было:





Стало:



Причина изменения:

Удалён метод-посредник handleClickWayHome, поскольку он просто вызывал showWayHome(teamId).

Упрощена логика кода, уменьшена вложенность и количество посредников.

Технический долг в коде Form.tsx проекта можно оценить как средний. В коде нет серьёзных проблем, но в будущем некоторые решения могли бы привести к сложностям при изменении функционала или масштабировании проекта. Избавление от посредников позволит сделать код более чистым. Создание собственных классов и хуков позволят повысить читаемость, поддержку кода, а также возможности переиспользования компонента.