|  |  |
| --- | --- |
|  | **Тестовый документ** |

## **Аннотация теста**

|  |  |
| --- | --- |
| **Название проекта** | REG\_MARK\_LIB |
| **Рабочая версия** | NET 8.0 |
| **Имя тестирующего** | Газизов Дамир |
| **Дата(ы) теста** | 01.03.2025 |

## **Расшифровка тестовых информационных полей:**

|  |  |
| --- | --- |
| **Поле** | **Описание** |
| **Название проекта** | Название тестируемого проекта |
| **Рабочая версия** | Версия проекта/программного обеспечения (первый тест считается 1.0). |
| **Имя тестирующего** | Имя того, кто проводил тесты |
| **Дата(ы) теста** | Дата(ы) проведения тестов – это один или несколько дней. Если тесты проводились в более протяженный период времени, нужно отметить отдельную дату для каждого теста. |
| **Тестовый пример #** | Уникальный ID для каждого тестового примера. Следуйте некоторым конвенциям, чтобы указать типы тестов. Например,‘TC\_UI\_1′ означает‘user interface test case #1′ ( ТС\_ПИ\_1: тестовый случай пользовательского интерфейса#1) |
| **Приоритет тестирования**  *(Низкий/Средний/Высокий)* | Насколько важен каждый тест. Приоритет тестирования для бизнес-правил и функциональных тестовых случаев может быть средним или высоким, в то время как незначительные случаи пользовательского интерфейса могут иметь низкий приоритет. |
| **Заголовок/название теста** | Название тестового случая. Например, Подтвердите страницу авторизации с действительным именем пользователя и паролем. |
| **Краткое изложение теста** | Описание того, что должен достичь тест. |
| **Этапы теста** | Перечислите все этапы теста подробно. Запишите этапы теста в том порядке, в котором они должны быть реализованы. Предоставьте как можно больше подробностей и разъяснений. Пронумерованный список – хорошая идея. |
| **Тестовые данные** | Перечислите/опишите все тестовые данные, используемые для данного тестового случая. Так, фактические используемые входные данные можно отслеживать по результатам тестирования. Например, Имя пользователя и пароль для подтверждения входа. |
| **Ожидаемый результат** | Каким должен быть вывод системы после выполнения теста? Подробно опишите ожидаемый результат, включая все сообщения/ошибки, которые должны отображаться на экране. |
| **Фактический результат** | Каким должен быть фактический результат после выполнения теста? Опишите любое релевантное поведение системы после выполнения теста. |
| **Предварительное условие** | Любые предварительные условия, которые должны быть выполнены до выполнения теста. Перечислите все предварительные условия для выполнения этого тестового случая. |
| **Постусловие** | Каким должно быть состояние системы после выполнения теста? |
| **Статус**  *(Зачет/Незачет)* | Если фактический результат не соответствует ожидаемому результату, отметьте тест как неудачный. В ином случае обновление пройдено. |
| **Примечания/комментарии** | Используйте эту область для любых дополнительных заметок/комментариев/вопросов. Эта область предназначена для поддержки вышеуказанных полей (например, если есть некоторые особые условия, которые не могут быть описаны в любом из вышеуказанных полей, или если есть вопросы, связанные с ожидаемыми или фактическими результатами). |

## Тестовый пример #1:

|  |  |
| --- | --- |
| **Тестовый пример #** | 1 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Проверка корректности первой буквы знака |
| **Краткое изложение теста** | Проверить, что метод возвращает true для допустимых первых букв знаков. |
| **Этапы теста** | * Подготовить тестовые данные с допустимыми первыми буквами (например, "A", "B", "C"). * Вызвать метод проверки первой буквы для каждого знака. * Зафиксировать результаты. |
| **Тестовые данные** | "A123BC456", "B123BC456", "C123BC456" |
| **Ожидаемый результат** | Метод должен вернуть true для всех тестовых данных. |
| **Фактический результат** | Метод возвращает true для всех тестовых данных. |
| **Статус** | Зачет |
| **Предварительное условие** | Класс Class1 должен быть инициализирован. |
| **Постусловие** | Результаты теста должны быть зафиксированы для анализа. |
| **Примечания/комментарии** | Код должен быть написан, без ошибок, с доступом ко всем функциям и переменным |

## Тестовый пример #2:

|  |  |
| --- | --- |
| **Тестовый пример #** | 2 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Проверка обработки пустой строки |
| **Краткое изложение теста** | Проверить, что метод выбрасывает исключение при передаче пустой строки. |
| **Этапы теста** | * Подготовить пустую строку как входные данные. * Вызвать метод проверки с пустой строкой. * Зафиксировать результат и проверить, выбрасывается ли исключение. |
| **Тестовые данные** | "" |
| **Ожидаемый результат** | Метод должен выбросить ArgumentException. |
| **Фактический результат** | Метод выбрасывает ArgumentException. |
| **Статус** | Зачет |
| **Предварительное условие** | Класс Class1 должен быть инициализирован. |
| **Постусловие** | Исключение должно быть зафиксировано. |
| **Примечания/комментарии** | Код должен быть написан, без ошибок, с доступом ко всем функциям и переменным |

## Тестовый пример #3:

|  |  |
| --- | --- |
| Тестовый пример **#** | 3 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Проверка инкрементации второго символа |
| **Краткое изложение теста** | Проверить, что метод корректно инкрементирует второй символ знака. |
| **Этапы теста** | * Подготовить тестовые данные со знаком "A123BC456". * Вызвать метод инкрементации. * Зафиксировать результат. |
| **Тестовые данные** | "A123BC456" |
| **Ожидаемый результат** | Метод должен вернуть "A123BC457". |
| **Фактический результат** | Метод возвращает "A123BC457". |
| **Статус** | Зачет |
| **Предварительное условие** | Класс Class1 должен быть инициализирован. |
| **Постусловие** | Результаты теста должны быть зафиксированы для анализа. |
| **Примечания/комментарии** | Код должен быть написан, без ошибок, с доступом ко всем функциям и переменным |

## Тестовый пример #4:

|  |  |
| --- | --- |
| Тестовый пример **#** | 4 |
| **Приоритет тестирования** | Средний |
| **Заголовок/название теста** | Проверка обработки некорректного знака |
| **Краткое изложение теста** | Проверить, что метод возвращает ошибку при передаче некорректного знака. |
| **Этапы теста** | * Подготовить тестовые данные с некорректным знаком (например, "Z123BC456"). * Вызвать метод обработки знака. * Зафиксировать результат и проверить, возвращается ли ошибка. |
| **Тестовые данные** | "Z123BC456" |
| **Ожидаемый результат** | Метод должен вернуть false или ошибку о некорректном знаке. |
| **Фактический результат** | Метод возвращает false или ошибку о некорректном знаке. |
| **Статус** | Зачет |
| **Предварительное условие** | Класс Class1 должен быть инициализирован. |
| **Постусловие** | Ошибка должна быть зафиксирована. |
| **Примечания/комментарии** | Код должен быть написан, без ошибок, с доступом ко всем функциям и переменным |

## Test case #5:

|  |  |
| --- | --- |
| **Test Case #** | 5 |
| **Приоритет тестирования** | Низкий |
| **Заголовок/название теста** | Проверка пользовательского интерфейса для ввода знака |
| **Краткое изложение теста** | Убедиться, что пользовательский интерфейс позволяет вводить знак корректно. |
| **Этапы теста** | * Открыть интерфейс приложения. * Ввести знак в соответствующее поле. * Нажать кнопку "Подтвердить". * Зафиксировать результат. |
| **Тестовые данные** | "A123BC456" |
| **Ожидаемый результат** | Знак должен быть успешно принят, и появится сообщение об успешной обработке. |
| **Фактический результат** | Знак был успешно принят, и появится сообщение об успешной обработке. |
| **Статус** | Зачет |
| **Предварительное условие** | Интерфейс приложения должен быть доступен. |
| **Постусловие** | Результаты ввода должны быть зафиксированы. |
| **Примечания/комментарии** | Код должен быть написан, без ошибок, с доступом ко всем функциям и переменным |