|  |  |  |
| --- | --- | --- |
|  | Изображение выглядит как текст, вычерчивание линий  Автоматически созданное описание |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий (ИТ)

Кафедра Математического обеспечения и стандартизации информационных технологий (МОСИТ)

**ПРАКТИЧЕСКАЯ РАБОТА №2**

по дисциплине

«Тестирование и верификация программного обеспечения»

**Тема: «Модульное тестирование ПО»**

**Студент группы** ИКБО-03-20 Голиков А.Л.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель практической работы**  Овчинникова М.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Практические работы выполнены «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Зачтено «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Москва 2022

# Содержание

[1 Цель работы 3](#_Toc115285248)

[1.1 Задание для выполнения 3](#_Toc115285249)

[1.2 Материальная часть 3](#_Toc115285250)

[2 Описание продукта 4](#_Toc115285251)

[3 Ход работы 5](#_Toc115285252)

[3.1 Анализ и уточнение требований 5](#_Toc115285253)

[3.2 Планирование тестирования 5](#_Toc115285254)

[3.3 Разработка тестов 5](#_Toc115285255)

[3.4 Выполнение тестов 7](#_Toc115285256)

[3.5 Чек-листы 7](#_Toc115285257)

[3.6 Оценка результатов 9](#_Toc115285258)

[4 Выводы 10](#_Toc115285259)

[5 Список использованных источников 11](#_Toc115285260)

# **Цель работы**

## **Задание для выполнения**

Изучение процесса тестирования программного обеспечения с помощью модульного тестирования.

## **Материальная часть**

Модульное тестирование, иногда блочное тестирование или юнит-тестирование (англ. unit testing) — процесс в программировании, позволяющий проверить на корректность отдельные модули исходного кода программы, наборы из одного или более программных модулей вместе с соответствующими управляющими данными, процедурами использования и обработки.

# **Описание продукта**

«Гороскопус» - программный продукт для отображения гороскопов для заданного пользователем, знака зодиака или общий гороскоп на день. Приложение использует данные с сайта, на котором каждый день публикуются гороскопы, как и к каждому знаку зодиака, так и общий гороскоп на день.

Среда разработки Visual Studio Code

Взаимодействие с программным продуктом производится с помощью ввода данных в консоль, программа имеет текстовый интерфейс и взаимодействует с пользователем, путем вывода в консоль текстовой информации.

# **Ход работы**

## **Анализ и уточнение требований**

**Характеристики пользователя:** пользователь способен выбрать знак зодиака, для которого будет выведен гороскоп, а также увидеть общий гороскоп на день.

**Требования продукта**: ПК/ноутбук с установленной на нем средой разработки, которая поддерживает язык программирования Python, библиотеки beautifulsoup, requests, tkinter, а также доступ к сети интернет.

## **Планирование тестирования**

Тест-план:

1. Список функций и компонентов тестируемой системы:
2. Стратегия тестирования: написание юнит-тестов с помощью библиотеки unittest.
3. Тестовые окружения: ноутбук, среда разработки, библиотека unittest, интернет.
4. Последовательность проведения работ: подготовка, тестирование, анализ результатов, учёт зависимостей тестовых активностей от задач разработки и смежных групп.
5. Риски и стратегии по их разрешению: риск - отсутствие необходимых требований к системе, решение — поиск необходимого функционала.

## **Разработка тестов**

После прочтения документации и анализа кода были написаны 4 юнит-теста.

Первый тест предназначен для проверки корректности данных массива с английскими названиями знака зодиака, правильные данные в этом массиве необходимы для корректного запроса приложения (см. листинг 1).

*Листинг 1. Первый юнит-тест*

class TestHoroscope(unittest.TestCase):

  def test\_SignsEn(self):

    self.assertEqual(printSignEn(3), "cancer")

Второй тест предназначен для проверки корректности данных массива с русскими и названиями знака зодиака, правильные данные в этом массиве необходимы для корректного вывода информации для пользователя приложения (см. листинг 2).

*Листинг 2. Второй юнит-тест*

class TestHoroscope(unittest.TestCase):

  def test\_SignsRu(self):

    self.assertEqual(printSignRu(4), "Львов")

Третий тест предназначен для тестирования функции, которая выводит общий гороскоп на день (см. листинг 3).

*Листинг 3. Третий юнит-тест*

class TestHoroscope(unittest.TestCase):

    def test\_ShowGeneral(self):

        self.assertEqual(showGeneral(), "3-4 лунные сутки. Внутренняя чуйка работает бесперебойно. Так что прислушиваемся к ней и принимаем только верные решения. Отмахиваемся от негативных мыслей, которые сегодня то и дело лезут в голову, поднимаем себе настроение и смотрим в будущее – оно точно будет светлым. Не стоит высовываться – сейчас лучше не маячить как на работе перед начальством, так и дома перед близкими и любимыми, чтобы на нас не вылились проблемы и претензии. А вот кошелек лучше лишний раз не открывать – можем потратить много денег, сами того не осознавая. Читайте личный гороскоп и не погружайтесь в прошлое.")

Четвертый тест предназначен для тестирования функции, которая выводит дневной гороскоп по знаку, введенному пользователем (см. листинг 4).

*Листинг 4. Четвертый юнит-тест*

class TestHoroscope(unittest.TestCase):

    def test\_showSign(self):

        self.assertEqual(showSign(3), "Внутренняя чуйка будет вести вас по правильному пути, поэтому доверяйте себе, особенно в самых сложных ситуациях. На время придется замереть – так проблемы вас не заметят и пройдут стороной. На работе тоже сильно не мелькайте, иначе вас загрузят делами по самое горло. И не берите работу на дом – в этот вечер звезды прописывают вам отдых в кругу семьи.")

## **Выполнение тестов**

Найденные ошибки: программный продукт успешно прошел все тесты, кроме второго. Ошибка показана на рисунке 1:

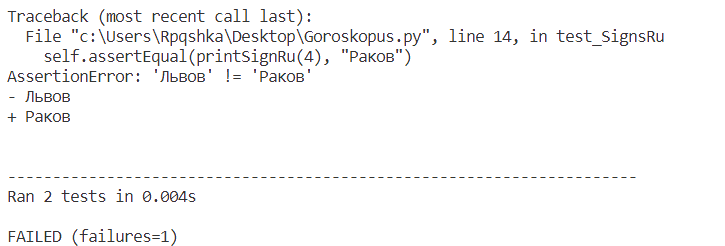


Рисунок 1 – пример ошибки при unit-тестировании

Замечания в ходе изучения документации: в документации не были описаны нужные компоненты для работы приложения, из-за чего программный продукт мог работать некорректно.

## **Чек-листы**

*Таблица 1 – Результаты тестирования*

|  |  |  |
| --- | --- | --- |
| **Входные значения** | **Ожидаемый результат** | **Полученные значения** |
| 3 | cancer | Соответствуют ожидаемым. |
| 4 | Раков | Львов |
| 1 | Внутренняя чуйка работает бесперебойно. Так что прислушиваемся к ней и принимаем только верные решения. Отмахиваемся от негативных мыслей, которые сегодня то и дело лезут в голову, поднимаем себе настроение и смотрим в будущее – оно точно будет светлым. Не стоит высовываться – сейчас лучше не маячить как на работе перед начальством, так и дома перед близкими и любимыми, чтобы на нас не вылились проблемы и претензии.  А вот кошелек лучше лишний раз не открывать – можем потратить много денег, сами того не осознавая. Читайте личный гороскоп и не погружайтесь в прошлое. | Соответствуют ожидаемым. |
| 2 | Внутренняя чуйка будет вести вас по правильному пути, поэтому доверяйте себе, особенно в самых сложных ситуациях. На время придется замереть – так проблемы вас не заметят и пройдут стороной. На работе тоже сильно не мелькайте, иначе вас загрузят делами по самое горло. И не берите работу на дом – в этот вечер звезды прописывают вам отдых в кругу семьи. | Соответствуют ожидаемым. |

## **Оценка результатов**

В ходе модульного тестирования была выявлена корректная работа всех тестов. кроме второго. Ошибка была вызвана тем, что в массиве были некорректные данные.

# **Выводы**

В ходе выполнения работы был протестирован проект «Гороскопус». Тестирование происходило методом модульного тестирования, был составлен план тестирования, проверены правила написания документации. Тестирование прошло успешно, ошибка выявлена только в одном тесте из четырех.

# **Список использованных источников**

1. Модульное тестирование [Электронный ресурс] – 2022 – Режим доступа: https://ru.wikipedia.org/wiki/Модульное\_тестирование.
2. Лекции учебного курса «Тестирование и верификация программного обеспечения».