



Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Информационная безопасность»

ОТЧЕТ

по лабораторной работе № 4

по курсу «Автодокументация в Python (Sphinx)»

на тему: «Реализовать систему автодокументации для моделей данных»

Студент ИУ8-13М
(Группа)

(Подпись, дата)

Савватеев А. Э.
(И. О. Фамилия)

Преподаватель

(Подпись, дата)

Зотов М. В.
(И. О. Фамилия)

2025 г.

СОДЕРЖАНИЕ

1	ЦЕЛЬ РАБОТЫ	3
2	ТРЕБОВАНИЯ ЛАБОРАТОРНОЙ РАБОТЫ	4
3	РЕАЛИЗОВАННАЯ РАБОТА	5
3.1	Документирование классов	5
4	ЗАКЛЮЧЕНИЕ	6

1 ЦЕЛЬ РАБОТЫ

Реализовать систему автодокументации для моделей данных с использованием `tox` + GitHub Actions.

2 ТРЕБОВАНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

- Описаны классы, методы и атрибуты в одном из популярных форматов (Google, NumPy или Sphinx)
- Добавлен GitHub Action, который запускает сборку документации по пути лабораторной работы
- Добавлена возможность запуска документации через `tox`

3 РЕАЛИЗОВАННАЯ РАБОТА

3.1 Документирование классов

В файле `students_folder/savandr/lab_2/lab_2.py` реализованы Pydantic модели с подробной документацией в формате Google docstrings:

- **UserSpec** — модель профиля пользователя с полями: `user_id`, `username`, `surname`, `second_name`, `email`, `status`. Включает валидацию email и проверку русских символов.
- **ProfileSpec** — расширенная модель пользователя, наследуется от **UserSpec**, добавляет поля `bio` и `url`.
- **ItemSpec** — модель товара с валидацией цены (должна быть > 0).
- **ServiceSpec** — модель услуги с аналогичной валидацией.
- **OrderLineSpec** — модель строки заказа с вычисляемым полем `line_price`.
- **OrderSpec** — модель заказа с профилем пользователя и списком строк.
- **OrdersSpec** — модель списка заказов маркетплейса.

Все классы содержат:

- Подробные docstring с описанием атрибутов
- Примеры использования в docstring
- Описание валидаторов и их логики
- Конфигурацию `extra='forbid'`
- Валидаторы через `@field_validator` и `@model_validator`

4 ЗАКЛЮЧЕНИЕ

Таким образом было сделано:

- Качественно документированные Pydantic модели с подробными Google docstrings
- Все классы содержат описание атрибутов, валидаторов и примеры использования
- Реализована валидация данных через Pydantic v2