|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство образования и науки Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ\_\_\_\_\_\_\_\_\_\_\_\_Информационное\_\_управление\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ ПО УЧЕБНОМУ ПРАКТИКУМУ**

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*подпись, дата фамилия, и.о.*

Ментор команды **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*подпись, дата фамилия, и.о.*

Руководитель практики **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*подпись, дата фамилия, и.о.*

*20 г.*

Оглавление

[Введение 3](#_Toc9361003)

[Аналитический раздел 4](#_Toc9361004)

[Конструкторский раздел 5](#_Toc9361005)

[Технологический раздел 6](#_Toc9361006)

[Личный вклад 7](#_Toc9361007)

[Заключение 8](#_Toc9361008)

[Литература 9](#_Toc9361009)

Введение

Во введении формулируется задача/проблема, решаемая командой во время практики; описывает роль, закрепленная за каждым участником команды.

Аналитический раздел

* Обзор существующих решений.
* Анализ их достоинств и недостатков.
* Обоснование необходимости нового решения (поскольку практика время проведения практики ограничено, она имеет учебный характер, этот пункт может отсутствовать).
* Постановка задачи.
* Выводы.

Конструкторский раздел

* Декомпозиция задачи.
* Задачи, поставленные перед каждым участником.
* Разработка структуры создаваемого программного продукта.
* Описание назначения, требований к выделенным компонентам и их интерфейса.
* Разработка алгоритмов и структур данных для выделенных компонент.
* Проектирование пользовательского интерфейса (в случае его нетривиальности).
* Описание способов тестирования как выделенных компонент, так и программного продукта в целом.
* Подготовка тестовых данных.
* Выводы.

Технологический раздел

* Выбор и обоснование технических средств (язык, БД, библиотеки).
* Выбор и обоснование модели разработки.
* Организация выпуска сборок, использование CI.
* Реализация программного продукта.
* Реализация тестирования: перечислить виды использованного тестирования (модульное/функциональное/интеграционное и т.п.), выбранные инструменты и привести количественные характеристики (например, сколько тестовых случаев в том или ином сценарии, величина покрытия кода тестами и т.п.).
* Развертывание разработанного программного продукта, инструкция для системного администратора (установка) и пользователя (использование).
* Выводы.

Личный вклад

Поставленные задачи:

* Создать модель пользователя
* Реализовать регистрацию новых пользователей
* Реализовать авторизацию существующих пользователей
* Реализовать механизм пользователем добавления “избранных” рецептов

Основной недостаток стандартной модели пользователя заключался в невозможности ее редактирования. Новая модель пользователя была названа Profile и была унаследована от стандартной Django модели User. Она была модифицирована: теперь есть возможность “привязывать” к ней избранные рецепты.

Регистрация новых пользователей происходит через стандартную форму UserCreationForm. В ней уже заложены всевозможные такие проверки на корректность введенных данных как:

* проверка, не зарегистрирован ли пользователь с таким именем
* проверка качества пароля
* Проверка на то, не совпадают ли имя пользователя и пароль
* Не состоит ли пароль только из цифр

проверка, не зарегистрирован ли пользователь с таким именем и т.д. Основной механизм описан в функции registration. Если форма валидна, то данные будут отправлены в базу данных, в противном случае пользователю сообщат об ошибке.

Авториция пользователей происходит с помощью встроенной функции auth, которая умеет хорошо работать с базой данных. Она проверяет, находится ли в базе данных пользователь с введенным ником и паролем. В случае, когда пользователь найден, она выполнит его авторизацию, в противном – сообщит пользователю об ошибке.

Добавление рецепта в избранное происходит с помощью привязки рецепта к пользователю связью рецепта и пользователя. Для этого в модифицированной модели пользователя необходимо свойство Favourites, в котором будут хранится рецепты. В дальнейшем в личном кабинете необходимо просто вывести множество рецептов, привязанных к конкретному пользователю.

Заключение

В заключении выполняется анализ успешности проектирования и реализации программного продукта, правильность выбранных решений и т.п. В случае если какой-то шаг был неуспешным, необходимо описать причины ошибки.

Литература