­­­­­­­­­ **Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет приложений»

Техническое задание

«Эволюция ближайших к Солнцу звезд»

| Выполнил: |  | Проверил: |
| --- | --- | --- |
| Студент группы ИУ5-54Б |  | Преподаватель каф. ИУ5 |
| Стукалов И.Д. |  | Канев А.И. |

Москва

2023 г.

СОДЕРЖАНИЕ

1. Цель
2. Назначение
3. Задачи
4. Методы веб-сервиса
5. Описание UI
6. Требования к аппаратному обеспечению

6.1. Сервер

6.2. Клиент

1. Требования к программному обеспечиванию

7.1. Серверные компоненты

7.2 Клиентские компоненты

1. **Цель**

Основная цель системы, состоящей из веб-сервиса и мобильного приложения, заключается в обеспечении ученых-астрономов актуальными данными о состоянии ближайших к Солнцу звезд. Данная система решает проблему мониторинга активности звезд и фиксировании происходящих с ними событий.

1. **Назначение**

Назначение приложения заключается в предоставлении информации о различных наблюдаемых звездах. Пользователи приложения смогут подавать заявки на события звезд, которые они указывают при редактировании заявки. Заявка на событие допускается к выполнению после подтверждения модератором. Модератор может принять заявку пользователя в работу или отклонить её. Пользователям доступен список их заявок, включающий активные и обработанные.

1. **Задачи**

3.1. Разработать дизайн системы (Golang templates).

3.2. Создать базу данных для хранения информации об операциях и заявках (PostgreSQL).

3.3. Создание веб-сервиса в бэкенде.

3.4. Написать интерфейс гостя (React-Vite).

3.5. Развертывание текущей версии на GitHub Pages.

3.6. Настроить авторизацию и Swagger в веб-сервисе.

3.7. Создание интерфейса для создания/просмотра заявки пользователем (React-Toolkit, Axios).

3.8. Создать нативное приложение, реализующее интерфейс гостя системы (React Native + Redux Toolkit).

3.9. Добавить асинхронный сервер для выполнения оплаты (Django).

3.10. Реализовать интерфейс модератора (React).

3.11. Составление документации (ТЗ, РПЗ и набор диаграмм).

1. **Методы веб-сервиса**

| **№** | **Описание** | **Метод** | **Path** | **Вход** | **Выход** |
| --- | --- | --- | --- | --- | --- |
| 4.1.1 | Возвращает список звезд | GET | /star | query params  {  name: CHARACTER VARYING,  dist\_top: INTEGER,  dist\_bot: INTEGER,  age\_top: INTEGER,  age:bot: INTEGER,  mag\_top: INTEGER,  mag\_bot: INTEGER,  } | {  draft\_id: INTEGER,  stars: [  {  star\_id: INTEGER,  name: CHARACTER VARYING,  description: CHARACTER VARYING,  distance: FLOAT,  age: FLOAT,  magnitude: FLOAT,  image: CHARACTER VARYING  }  ]  } |
| 4.1.2 | Возвращает одну звезду | GET | /star/<int:pk> | отсутствует | {  star\_id: INTEGER,  name: CHARACTER VARYING,  description: CHARACTER VARYING,  distance: FLOAT,  age: FLOAT,  magnitude: FLOAT,  image: CHARACTER VARYING  } |
| 4.1.3 | Добавляет одну звезду | POST | /star | multipart/form-data  {  name: CHARACTER VARYING,  description: CHARACTER VARYING,  distance: FLOAT,  age: FLOAT,  magnitude: FLOAT,  image: TEXT  } | {  message: TEXT  } |
| 4.1.4 | Удаляет одну звезду | PUT | /star/<int:pk>/delete | отсутствует | {  message: TEXT  } |
| 4.1.5 | Обновляет информацию о звезде | PUT | /star/<int:pk>/update | multipart/form-data  {  name: CHARACTER VARYING,  description: CHARACTER VARYING,  distance: FLOAT,  age: FLOAT,  magnitude: FLOAT,  image: CHARACTER VARYING  } | {  message: TEXT  } |
| 4.2.1 | Возвращает список событий звезд | GET | /event | query params  {  status: CHARACTER VARYING,  start\_formation: DATE,  end\_formation: DATE,  } | [  {  event\_id: INTEGER,  name: CHARACTER VARYING,  status: CHARACTER VARYING,  creation\_date: DATE,  formation\_date: DATE,  completion\_date: DATE,  moderator\_id: INTEGER,  creator\_id: INTEGER,  },  ] |
| 4.2.2 | Возвращает одно событие | GET | /event/<int:pk> | отсутствует | {  event\_id: INTEGER,  name: CHARACTER VARYING,  status: CHARACTER VARYING,  creation\_date: DATE,  formation\_date: DATE,  completion\_date: DATE,  moderator\_id: INTEGER,  creator\_id: INTEGER,  }, |
| 4.2.3 | Обновляет информацию о событии | PUT | /event/<int:pk> | query params  {  name: CHARACTER VARYING,  }, | {  message: TEXT  } |
| 4.2.4 | Добавляет звезду в событие | PUT | /event/star | {  star\_id: INTEGER  } | {  message: TEXT  } |
| 4.2.5 | Удаляет событие | PUT | /event/delete | отсутствует | {  message: TEXT  } |
| 4.2.6 | Формирует событие | PUT | /event/form | отсутствует | {  message: TEXT  } |
| 4.2.7 | Изменяет статус заявки | PUT | /event/<int:pk>/status | query params  {  status: TEXT  } | {  message: TEXT  } |
| 4.2.8 | Удаляет звезду из события | DELETE | /star-event/<int:pk> | отсутствует | {  message: TEXT  } |

1. **Описание UI (функциональные требования)**

5.1. Главное меню

5.1.1. Звезды – переход страницу звезд.

5.1.2. Заявки — переход на страницу заявок

*Для авторизованных пользователей*

5.1.3. Корзина — переход на страницу несформированной заявки (черновика)

*Для авторизованных пользователей с несформированной заявкой*

5.1.4. Вход – переход на страницу с формой авторизации.

*Только для неавторизированных пользователей.*

5.1.5. Выход – выход из аккаунта и переход на страницу звезд в режиме гостя.

*Для авторизованных пользователей.*

5.2. Окно авторизации.

*Доступно только неавторизованным пользователям.*

5.2.1. Ввод логина и пароля пользователя.

5.2.2. Кнопка «Войти». При нажатии на кнопку «Войти» выполняется авторизация пользователя. Если авторизация прошла успешно, пользователь перенаправляется на страницу списка звезд. Если введены неверные данные, то всплывает сообщение об ошибке.

5.2.3. Кнопка «Регистрация». При нажатии на кнопку «Регистрация» переходит на страницу регистрации.

5.2.4. Кнопка «Закрыть/Отменить». При нажатии на кнопку «Закрыть/Отменить» переходит к списку звезд.

5.3. Страница регистрации.

*Доступно только неавторизованным пользователям*

5.3.1. Ввод логина и пароля пользователя.

5.3.2. Кнопка «Зарегистрироваться». При нажатии на кнопку «Войти» выполняется регистрация пользователя. Если регистрация прошла успешно, пользователь перенаправляется на страницу списка звезд. Если введены некорректные данные, то всплывает сообщение об ошибке.

5.3.3. Кнопка «Войти». При нажатии на кнопку «Войти» переходит на страницу авторизации.

5.3.4. Кнопка «Закрыть/Отменить». При нажатии на кнопку «Закрыть/Отменить» переходит к списку звезд.

**6. Требования к аппаратному обеспечению**

6.1. Сервер

6.1.1. Процессор: минимум 2-ядерный

6.1.2. Оперативная память: минимум 4 ГБ

6.1.3. Доступ к интернету

**7. Требования к программному обеспечению**

7.1. Серверные компоненты:

7.1.1. Веб-браузер с поддержкой HTML5 и CSS3

7.1.2. Golang 1.20

7.1.3. PostgreSQL

7.1.4. Python с фреймворком Django

7.1.5. Node.js

7.1.6. Redis

7.1.7. Minio

7.2. Клиентские компоненты:

7.2.1. FireFox Browser 120.0 (64-разрядный)