

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ
УНИВЕРСИТЕТ ИТМО

Факультет инфокоммуникационных технологий

Образовательная программа 09.03.03

Направление подготовки (специальность) Мобильные сетевые технологии

О Т Ч Е Т

о курсовой работе

Тема задания: реализация клиентской части сервиса поиска прогноза погоды
средствами фреймворка Vue.JS

Обучающийся Костылев Иван Михайлович, К33401

Руководитель: Добряков Д. И., преподаватель

Оценка за курсовую работу ____

Подписи членов комиссии:

____ (Добряков Д.И.)
(подпись)

Дата ____

Санкт-Петербург
2021

ВВЕДЕНИЕ	3
Актуальность	3
Цели и задачи	3
ГЛАВА 1. СРЕДСТВА РАЗРАБОТКИ И ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ	4
1.1 Средства разработки	4
1.2 Функциональные требования	5
ГЛАВА 2. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ	6
2.1. Проектирование и реализация моделей	6
2.2. Реализация сериализаторов	10
2.3. Реализация представлений	11
ГЛАВА 3. ИССЛЕДОВАНИЕ	17
3.1. Начало работы	17
3.2. Дать django возможность работать с уже существующей БД	17
3.3. Выбор метрик	17
3.4. Фильтруем в django	18
3.4.1. Contains	19
3.4.2. Icontains	21
3.4.3. Full text search (через django.contrib.postgres)	22
3.4.4. Full text search (через rest_framework.filters, точнее — SearchFilter)	25
3.4.5. Использование фильтров через модуль django-filter	27
ЗАКЛЮЧЕНИЕ	28
Выводы по исследованию	28
Выводы по работе	28
СПИСОК ЛИТЕРАТУРЫ	29

ВВЕДЕНИЕ

Актуальность

Электронные сервисы для поиска прогноза погоды давно получили популярность. Это легкий способ получить информацию о состоянии погоды на улице, что является особенно важно в наше время, когда у нас нет возможности, допустим, возвращаться домой, чтобы надеть утепленную одежду. В ежедневной рутине люди хотят получать нужную им информацию быстро и легко. Наш сервис как раз-таки поможет это сделать.

Цели и задачи

1. Определение средств разработки
2. Определение функциональных требований
3. Проектирование и реализация клиентского приложения

ГЛАВА 1. СРЕДСТВА РАЗРАБОТКИ И ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ

1.1 Средства разработки

Большинство современных веб-клиентов основаны на трех основных паттернах - MPA, SPA и PWA. Каждый из них удобен для реализации сайтов различных типов. SPA, как правило, предназначен для отображения небольшого количества информации. Для реализации сервиса по поиску прогноза погоды он подходит очень хорошо.

Выбирая среди фреймворков для реализации данного паттерна, взгляд бросается Vue.JS. Для того, чтобы начать разрабатывать на нем, нужен низкий порог вхождения, и его документация достаточно подробна и исчерпывающая. При этом его возможностей и инструментария достаточно для разработки даже достаточно сложных приложений.

1.2 Функциональные требования

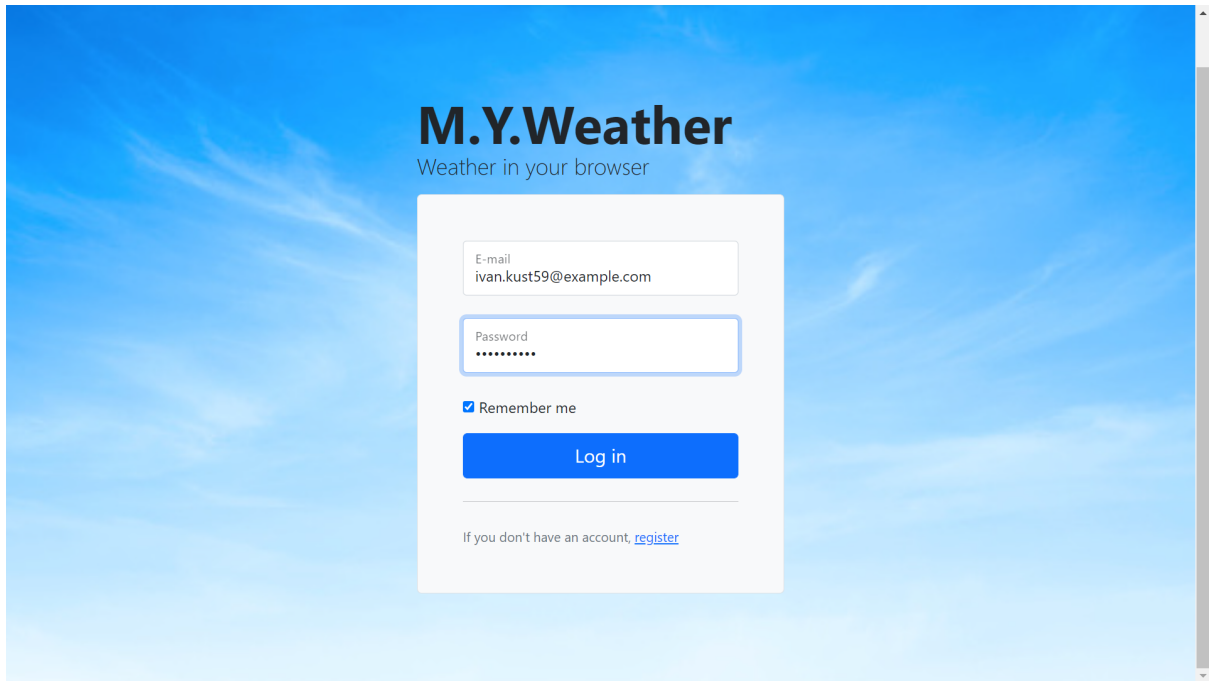
Функциональные требования к этому проекту выражаются в списке экранов, которые необходимо было реализовать:

1. Интерфейс регистрации;
2. Форма входа в аккаунт;
3. Интерфейс отображения погоды по выбранным городам;
4. Интерфейс поиска по городам;
5. Настройки пользователя, аккаунта;
6. Подробный прогноз погоды.

ГЛАВА 2. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ

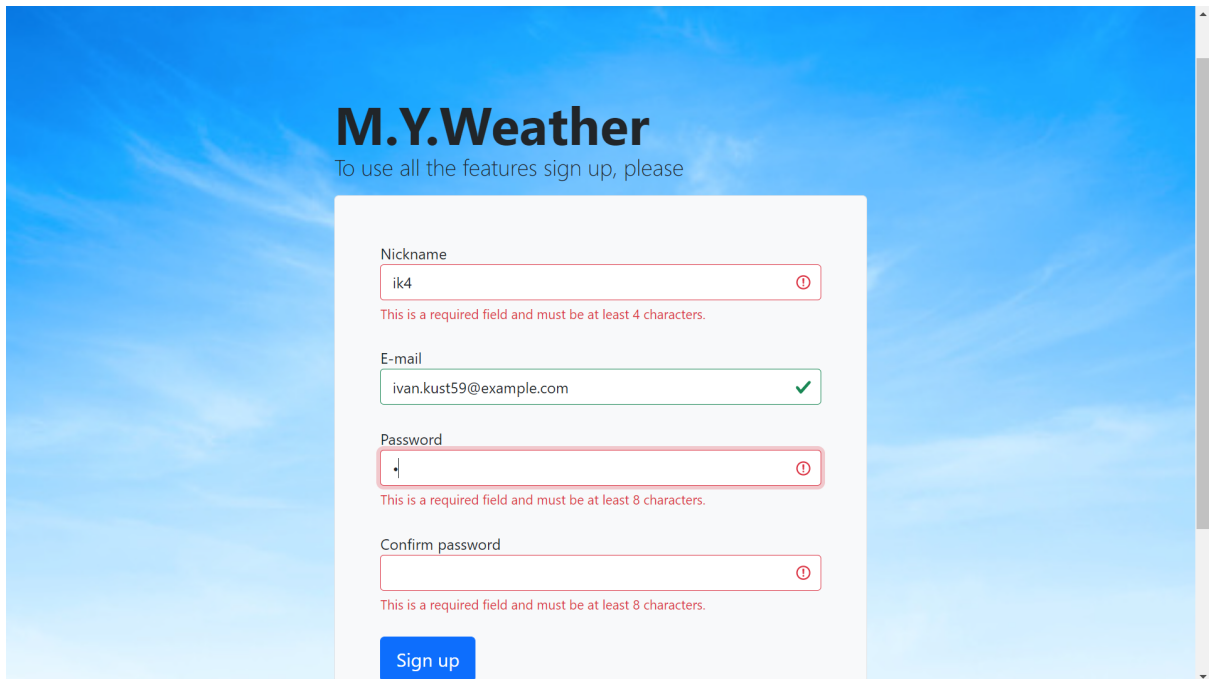
В рамках работы было реализовано 4 экрана

Экран входа.



The screenshot shows the login interface for 'M.Y. Weather'. The background is a blue sky with white clouds. The title 'M.Y. Weather' is prominently displayed, followed by the tagline 'Weather in your browser'. The login form is centered and contains an email input field with the text 'ivan.kust59@example.com', a password input field with masked characters, a checked 'Remember me' checkbox, and a blue 'Log in' button. Below the button is a link to 'register' for users who do not have an account.

Экран регистрации.




The screenshot shows the registration interface for 'M.Y. Weather'. The background is a blue sky with white clouds. The title 'M.Y. Weather' is displayed, followed by the instruction 'To use all the features sign up, please'. The registration form is centered and includes fields for 'Nickname' (containing 'ik4' with a red error message: 'This is a required field and must be at least 4 characters.'), 'E-mail' (containing 'ivan.kust59@example.com' with a green checkmark), 'Password' (containing a single character with a red error message: 'This is a required field and must be at least 8 characters.'), and 'Confirm password' (empty with a red error message: 'This is a required field and must be at least 8 characters.'). A blue 'Sign up' button is at the bottom.

Главная страница


M.Y.Weather Main Search Log out About authors

Hello, Ivan!




Saint Petersburg

Date	Temp	Prec
today	-8.58C	broken clouds
tomorrow	-4.21C	light snow
29.01	-3.15C	light snow



Moscow

Date	Temp	Prec
today	-5.32C	snow
tomorrow	-9.48C	overcast clouds
29.01	-8.08C	light snow




Perm

Date	Temp	Prec
today	-13.66C	scattered clouds
tomorrow	-9.47C	light snow
29.01	-11.13C	overcast clouds

Экран поиска

M.Y.Weather Main Search Log out About authors

Search weather for town



Ivanovo

-5°C

19.01	-3°C	snowing
20.01	-6°C	no precipitation
21.01	+2°C	rain
22.01	-6°C	no precipitation
23.01	-2°C	no precipitation
24.01	-3°C	no precipitation

Объявленные пути:

```

Vue.use(VueRouter)

const routes = [
  {
    path: '/',
    name: 'Weather',
    component: Weather
  },
  {
    path: '/signin',
    name: 'SignIn',
    component: SignIn
  },
  {
    path: '/signup',
    name: 'SignUp',
    component: SignUp
  },
  {
    path: '/search',
    name: 'Search',
    component: Search
  }
]

```

Базовый компонент App.vue содержит навигационную панель на всех страницах

```

<template>
  <div id="app">
    <NavBar/>
    <router-view/>
  </div>
</template>

<script>
import NavBar from '@components/NavBar'

export default {
  components: { NavBar }
}
</script>

```

Запросы к API выполняются с помощью axios

```
mounted () {  
  axios  
    .get('https://api.openweathermap.org/data/2.5/onecall?lat=' + MOSCOW.lat + '&lon=' + MOSCOW.lon + '&exclude={d  
    .then(response => response.data)  
    .then(data => {  
      weather2 = [  
        { date: 'today', temp: data.daily[0].temp.day + 'C', prec: data.daily[0].weather[0].description },  
        { date: 'tomorrow', temp: data.daily[1].temp.day + 'C', prec: data.daily[1].weather[0].description },  
        { date: '29.01', temp: data.daily[2].temp.day + 'C', prec: data.daily[2].weather[0].description }  
      ]  
    })  
}
```

Верстка компонентов была выполнена с использованием Bootstrap-Vue

```
import Vue from 'vue'  
import { BootstrapVue, IconsPlugin } from 'bootstrap-vue'  
import App from './App.vue'  
import router from './router'  
  
import 'bootstrap/dist/css/bootstrap.css'  
import 'bootstrap-vue/dist/bootstrap-vue.css'  
  
Vue.config.productionTip = false  
  
new Vue({  
  router,  
  render: h => h(App)  
}).$mount('#app')  
  
Vue.use(BootstrapVue)  
Vue.use(IconsPlugin)
```


ЗАКЛЮЧЕНИЕ

Выводы по работе

Благодаря этой работе, мы овладели базовыми навыками работы над клиентским веб-приложением. Использование инструментов фреймворка Vue.JS оказалось достаточно удобным для реализации. Фреймворк Vue.JS какими-то частями напоминает фреймворк Android - например, наличие роутеров или вложенные друг в друга компоненты, переиспользование виджетов.

СПИСОК ЛИТЕРАТУРЫ

1. Что такое MPA, SPA и PWA [Электронный ресурс] <https://www.azoft.ru/blog/spa-mpa-pwa/>
2. Документация Vuex [Электронный ресурс] <https://vuex.vuejs.org/ru/>
3. Документация Vue-router [Электронный ресурс] <https://router.vuejs.org/ru/api/>
4. Документация VueJs [Электронный ресурс] <https://vuejs.org/>