

## REACTJS VA VUEJS HAQIDA TUSHUNCHA VA ULAR ORASIDAGI FARQ

*To`xtayeva Robiya Ravshanbek qizi*

*Andijon davlat universiteti talabasi*

**Anotatsiya:** JavaScriptning ReactJs va VueJs veb-illovalarini ishlab chiqish uchun eng asosiy texnologiyalaridir. Ushbu maqolada React va Vue haqida to`liq malumot topib, orasidagi farqni ko`rib chiqamiz. Biz har birining ijobiy va salbiy tomonlari haqida gaplashamiz, qaysi birini tanlash bo'yicha maslahatlar beramiz.

**Kalit so`zlar:** ReactJs, Javascript, statistika, VueJs, funksiannallik, componentlar, web-sahifa, web-browser, DOM

Dasturlash deganda kompyuterga muammolarni hal qilish uchun qanday vazifalarni bajarish kerakligini aytish uchun texnologik jarayon tushuniladi. Siz dasturlashni odamlar va kompyuterlar o`rtasidagi hamkorlik deb o`ylashingiz mumkin, bunda odamlar kompyuterlar tushunadigan tilda kompyuterga rioya qilish (kod) uchun ko`rsatmalar yaratadilar.

Turli xil dasturlash tillari dasturchilarga kompyuterlar tushunadigan kod yozish imkonini beradi. Statista tomonidan o`tkazilgan so`rovga ko`ra, ishlab chiquvchilar foydalanadigan eng yaxshi beshta dasturlash tillari:

**JavaScript** dan 16,8% foydalaniladi

**C** dan 15,7% foydalaniladi

**Python** , 9,7% foydalanadi

**C++** , 5,5% tomonidan ishlatiladi

**C#**, 5,3% tomonidan foydalaniladi

Front-end dasturchi - bu veb-saytlar va veb-illovalarning foydalanuvchi interfeysi (UI) va foydalanuvchi tajribasini (UX) yaratish va loyihalashga ixtisoslashgan dasturiy ta'minot ishlab chiqaruvchisi. Front-end dasturchining asosiy mas'uliyati veb-sayt yoki ilovaning vizual va interaktiv jihatlari foydalanuvchilarga qulay, estetik jihatdan yoqimli va funktsional jihatdan samarali bo'lishini ta'minlashdir. Biz ko`proq front-end uchun Javascript dasturlash tilidan foydalanamiz.

JavaScript - bu veb-sahifalarda murakkab funksiyalarni amalga oshirish imkonini beruvchi skript yoki dasturlash tilidir - har safar veb-sahifa shunchaki yaratilmaydi, siz o`ylagandan ko`ra ko`proq narsalarni qiladi va siz ko`rishingiz uchun statik ma'lumotlarni ko`rsatadi - o`z vaqtida kontent yangilanishlarini, interaktiv xaritalarni, animatsion 2D/ 3D grafika, aylantiruvchi video roliklar va boshqalar — JavaScript-ni jalb qilish mumkinligiga ishonch hosil mumkin.

Bugungi kunda front-end ramkalari va kutubxonalar zamonaviy veb-ishlab chiqishning muhim qismiga aylanmoqda. Javascript uchun: Angular, VueJs, ReactJs larni misol qilsak bo`ladi.

React.js - JavaScript hamjamiyatida asta-sekin zamonaviy veb-ishlab chiqish uchun asosiy asosga aylangan frontend kutubxona. Veb-ishlab chiqishda yangi bo`lganlar yoki bu nima haqida ekanligini tushunishga harakat qilayotganlar uchun keling, ReactJsni, uning qanday ishlashini va uning boshqa JavaScript ramkalaridan nimasi bilan farq qilishini ko`rib chiqaylik. Bu kutubxona Facebook tomonidan aniqlik bilan yaratilgan ochiq manbali JavaScript kutubxonasi bo`lib, u interaktiv foydalanuvchi interfeyslarini yaratishning murakkab jarayonini soddalashtirishga qaratilgan. ReactJs bilan yaratilgan foydalanuvchi interfeysini har biri kichik, qayta foydalanish mumkin bo`lgan HTML kodini chiqarish uchun mas'ul bo`lgan komponentlar to`plami sifatida tasavvur qiling. Undan birinchi bo`lib Facebook yangiliklar lentasi foydalandi. DOM manipulyatsiyasi va foydalanuvchi interfeyslariga inqilobiy yondashuvi tufayli ReactJs Facebookning veb-ishlab chiqishga bo`lgan yondashuvini keskin o`zgartirdi va ochiq manbalar hamjamiyatiga chiqarilgandan so`ng JavaScript ekotizimida tezda mashhur bo`ldi.

React.js odatda, veb-sahifani veb-brauzeringizga URL manzilini kiritish orqali so`raysiz. Keyin brauzeringiz o`sha veb-sahifa uchun so`rov yuboradi, bu sizning brauzeringizda ko`rsatadi. Agar veb-saytdagi boshqa sahifaga o`tish uchun ushbu veb-sahifadagi havolani bossangiz, ushbu yangi sahifani olish uchun serverga yangi so`rov yuboriladi. Brauzeringiz (mijoz) va server o`rtasidagi bu oldinga va orqaga yuklash tartibi veb-saytga kirishga harakat qilayotgan har bir yangi sahifa yoki manba uchun davom etadi. Veb-saytlarni yuklashning bunday odatiy yondashuvi juda yaxshi ishlaydi, lekin juda ma'lumotlarga asoslangan veb-saytni ko`rib chiqing. To`liq veb-sahifani oldinga va orqaga yuklash ortiqcha bo`ladi va yomon foydalanuvchi tajribasini yaratadi. Bundan tashqari, an'anaviy JavaScript ilovasida ma'lumotlar o`zgarganda, bu o`zgarishlarni aks ettirish uchun qo`lda DOM manipulyatsiyasi talab qilinadi. Qaysi ma'lumotlar o`zgarganligini aniqlashingiz va ushbu o`zgarishlarni aks ettirish uchun DOMni yangilashingiz kerak, natijada sahifa to`liq qayta yuklanadi.

VueJs — foydalanuvchi interfeyslarini yaratish uchun JavaScript asosidir. U standart HTML, CSS va JavaScript-ga asoslanadi va oddiy yoki murakkab bo`lishidan qat'i nazar, foydalanuvchi interfeyslarini samarali rivojlantirishga yordam beradigan deklarativ va komponentlarga asoslangan dasturlash modelini taqdim etadi.

Mana minimal misol:

```
import { createApp, ref } from 'vue'

createApp({
  setup() {
    return {
      count: ref(0)
    }
  }
}).mount('#app')
```

```
<div id="app">
  <button @click="count++">
    Count is: {{ count }}
  </button>
</div>
```

Yuqoridagi misol Vue-ning ikkita asosiy xususiyatini ko'rsatadi:

**Deklarativ ko'rsatish :** Vue standart HTMLni shablon sintaksisi bilan kengaytiradi, bu bizga JavaScript holatiga asoslangan HTML chiqishini deklarativ tarzda tavsiflash imkonini beradi.

**Reaktivlik :** Vue avtomatik ravishda JavaScript holati o'zgarishlarini kuzatib boradi va o'zgarishlar sodir bo'lganda DOMni samarali yangilaydi. Sizda allaqachon savollar bo'lishi mumkin - tashvishlanmang. Qolgan hujjatlarda biz har bir kichik tafsilotni ko'rib chiqamiz. Hozircha, iltimos, Vue taklif qilayotgan narsalarni yuqori darajada tushunishingiz uchun o'qing.

Vue - frontend ishlab chiqishda zarur bo'lgan umumiy xususiyatlarning ko'pini qamrab oluvchi Javascript ramkasi va ekotizim. Ammo veb juda xilma-xildir - biz Internetda yaratadigan narsalar shakl va miqyosda keskin farq qilishi mumkin. Shuni yodda tutgan holda, Vue moslashuvchan va asta-sekin qabul qilinadigan bo'lishi uchun yaratilgan. Foydalanish holatlaringizga qarab, Vue turli yo'llar bilan ishlatilishi mumkin:

- Qurilish bosqichisiz statik HTMLni yaxshilash
- Har qanday sahifaga veb-komponent sifatida joylashtirish
- Bir sahifali dastur (SPA)
- Fullstack / Server tomonida ko'rsatish (SSR)
- Jamstack / Statik sayt yaratish (SSG)
- Maqsadli ish stoli, mobil, WebGL va hatto terminal

Agar siz ushbu tushunchalarni tushunmayotgan bo'lsangiz, tashvishlanmang! Qo'llanma faqat HTML va JavaScript-ning asosiy bilimlarini talab qiladi va siz ularning hech birida mutaxassis bo'lmasdan kuzatib borishingiz kerak. Moslashuvchanligiga qaramay, Vue qanday ishlashi haqidagi asosiy ma'lumotlar ushbu foydalanish holatlarining barchasida taqsimlanadi. Agar siz hozir yangi boshlovchi bo'lsangiz ham, kelajakda yanada ulug'vor maqsadlarga erishish uchun o'sib ulg'ayganingizda, yo'lda olingan bilim foydali bo'lib oladi. Agar siz bular haqida oldindan bilsabgiz, siz hal qilmoqchi bo'lgan muammolarga asoslanib, bir xil

samaradorlikni saqlab qolgan holda Vue-dan foydalanishning optimal usulini tanlashingiz mumkin. Shuning uchun biz Vue-ni "Progressive Framework" deb ataymiz: bu siz bilan birga rivojlana oladigan va ehtiyojlaringizga moslasha oladigan ramka. Yagona faylli komponentlar. Ko'pgina qurilish vositalarini qo'llab-quvvatlaydigan Vue loyihalarida biz Vue omponentlarini bitta fayl komponenti deb nomlangan HTML-ga o'xshash fayl formatidan foydalangan holda yozamiz. Vue SFC, nomidan ko'rinib turibdiki, komponentning mantig'ini (JavaScript), shablonni (HTML) va uslublarni (CSS) bitta faylga qamrab oladi. SFC formatida yozilgan oldingi misol:

```

<script setup>
import { ref } from 'vue'
const count = ref(0)
</script>

<template>
  <button @click="count++">Count is: {{ count }}</button>
</template>

<style scoped>
button {
  font-weight: bold;
}
</style>

```

**SFC Vue-ning o'ziga xos xususiyati bo'lib, agar** sizning foydalanish holatingiz qurilishni sozlashni talab qilsa, Vue komponentlarini yaratishning tavsiya etilgan usuli hisoblanadi. Siz SFC qanday va nima uchun haqida uning maxsus bo'limida ko'proq bilib olishingiz mumkin - ammo hozircha shuni bilingki, Vue siz uchun barcha qurish vositalarini sozlash bilan shug'ullanadi.

ReactJs va VueJsni taqqoslaylik. React va Vue o'rtasidagi asosiy farq, Vue JSX-dan tashqari HTML shablonlaridan foydalanadi, reaksiya esa JSX-dan alohida foydalanadi. React va Vue ni hisobga olsak, React kutubxonadir; boshqa tomondan, Vue - bu ramka. React komponentlar arxitekturasi, DOM manipulyatsiyasi va komponent holatini boshqarishga ega. Taqqoslash uchun, Vue ko'plab vositalar va kutubxonalar va ko'plab uchinchi tomon vositalariga ega. React JavaScript-ni to'liq tushunishni talab qiladi, Vue esa kodlash bo'yicha asosiy bilimlarni talab qiladi.

React JavaScript-ni ishlab chiqish dunyosida paydo bo'lganida, bu Angular 1, Backbone.js yoki jQuery kabi o'sha paytdagi mavjud ramkalar bilan solishtirganda biroz o'zgardi. Eng muhimi, React Virtual DOM-ga ommaviy bozorni jalb qildi va ishlab chiquvchilarga HTML-ni JavaScript-da yozishga yordam berish uchun JavaScript XML (JSX) deb nomlangan yangi sintaksisni taqdim etdi.

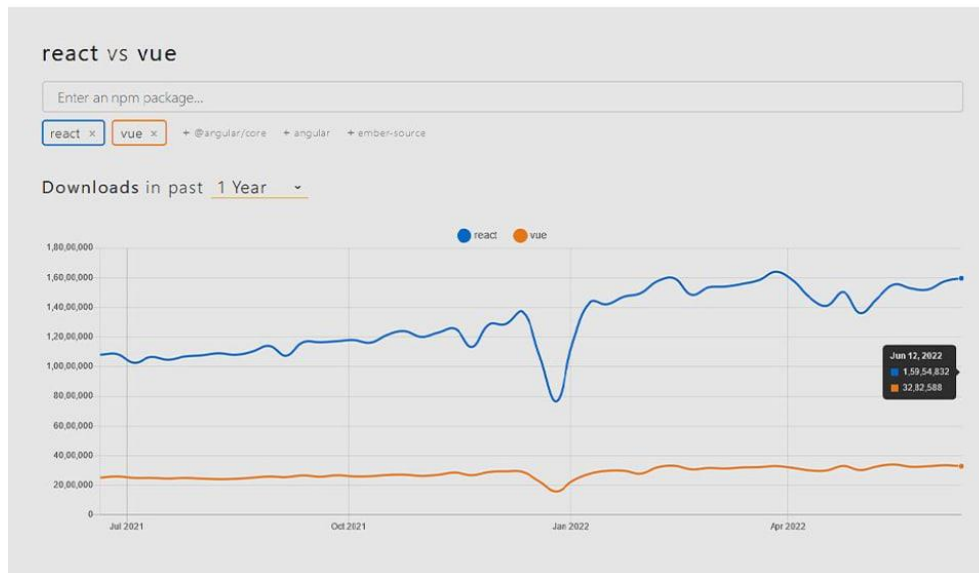
Hozirda ReactJS da jami 10 630 448 ta jonli veb-saytlar ishlaroqda.

Vue Evan You ismli sobiq Google dasturchisining qo'li bilan paydo bo'lgan.

Uning maqsadi yaxshilangan kodlash va qo‘shimcha funktsiyalar uchun Angular, Ember va Reactning eng yaxshilarini birlashtirgan ramka yaratish edi. Vue React bilan bir xil muammolarni hal qilishga qaratilgan, ammo boshqacha yo‘l bilan. U oddiy HTML-ga ega shablon tizimidan foydalanadi, siz uni mavjud kodga osongina qo‘shishingiz mumkin. Yirik korporatsiya o‘rniga Vue mustahkam ishlab chiquvchilar hamjamiyatiga ega bo‘lib, ular ushbu asosni oldinga siljitish uchun doimo qo‘llab-quvvatlaydigan va takomillashtirib boradi.

Hozirda 1,936,822 jonli veb-saytlar VueJS-dan foydalanmoqda.

React va Vue-dan yuklab olingan npm paketlar soni:



Loyihangiz uchun eng yaxshi frontend ramka yoki kutubxonasini tanlash juda qiyin, chunki u ko‘p sonli omillarga bog‘liq - loyihangiz ehtiyojlari, ramka xususiyatlari, ishlab chiquvchilarning mavjudligi va boshqalar. Bu bizni har bir mijozning buyurtmasi qaysi ramkaga mos kelishiga e‘tibor qaratishi kerakligiga ishonтиради.

Bundan tashqari, yakuniy tanlovingizga katta ta‘sir ko‘rsatishi mumkin bo‘lgan loyihaning murakkabligi va hajmi kabi boshqa o‘zgaruvchilar ham hisobga olinadi. React va Vue ikkalasining ham o‘ziga xos afzalliklari va kamchiliklari bor va ishlab chiquvchilaringizning tajribasi bu erda ham hal qiluvchi rol o‘ynaydi.

### Foydalanilgan adabiyotlar:

1. VueJs.org
2. ReactJs.org
3. “Pro VueJs” Adam Freeman kitobi
4. “Building vueJs” Heitor Ramon Roberio.
5. “VueJs 3 By example” John AU – Yeung
6. radixweb.com