

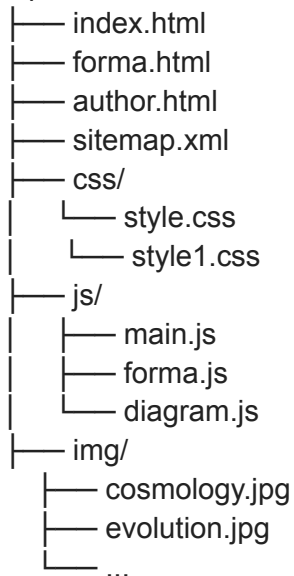
Visoka ICT Skola 2025

Dokumentacija

Web Programiranje 1

Struktura

/ExploreScience



Sadržaj po fajlovima

1. index.html

Glavna stranica (Home) - Kontejneri za kartice, heder, footer i modali, kartice modala i animacije u njima. Link ka formi za objavljivanje rezultata eksperimenta.

2. forma.html

Stranica koja sadrži formu za eksperiment. Validacija je u [main.js](#). Regexi za svaki element.

3.author.html

Stranica koja sadrži kratak opis autora plus link ka dokumentaciji.

4.Sitemap.xml

Sitemap za SEO.

5. style1.css

flex mode layout. ovaj dodatni fajl postoji jer sam planirao da postoji još jedna stranica "test equations" gde bi korisnici unosili razne podatke u polja i program bi prikazivao animirana rešenja tih jednačina. Nisam stigao da uradim ovu stranicu, ali sam planirao odvojen css zbog preglednosti. Iskoristio sam ga za formu.

6. style.css

flex mode. ovde je ceo css. I responsive na dnu. Flex i relativne vrednosti u formulama su omogućili da mi ne treba previse media za responsive.

7. JS

7.1. - [Main.js](#)

7.1.1. Inicijalizacija sadržaja

Definisana kolekcija topics ("const topics") – niz objekata gde svako ima:

title – naslov,

description – opis,

moreInfo – dodatni info,

img – slika za prikaz na kartici,

bglmg – pozadinska slika za modal,

diagramType – tip dijagrama za canvas

7.1.2. Generisanje kartica (glossary cards)

Za svaku temu kreira se div element sa klasom glossary_card.

U karticu se ubacuju dinamički: slika, naslov i kratak opis.

Klik otvara modalni prozor.

7.1.3. Modalni prozor

Kada se klikne, dinamički se kreira HTML za modal koji ima:

pozadinsku sliku,

naslov,

paragrafe,

<canvas> element za crtanje.

7.1.4, Dijagram

Funkcija `drawDiagram()` se poziva sa param `diagramType` kako bi iscrtao odgovarajući visual.

`bigBang` - ova animacija crta krug koji se siri. predstavlja ekspanziju vreme prostora.

`evol` - crta liniju koja se grana. Kao vrste po teoriji evolucije.

`elektron` - crta nukleus i elektronovu orbitu oko nukleusa. Po danasnoj kvantnoj mehanici, ovaj model prikaza je zastareo. Elektronske orbite se ovih dana vizuelno predstavljaju kao oblak potencijala oko nukleusa. Ali, crtati ovu animaciju je bilo pretesko, stoga stari model vodonika

`superPozicija` - crta linije koje treba da predstavljaju superpoziciju dve cestice. Nije ispalo tacno kako treba. Linije trebaju da nestanu nakon sto se spoje prvi put i onda ponovo da se odvoje i tako dalje. Ali ovo je bilo veoma komplikovano za postavljanje.\

`srce` - ovo je animacija koja crta srce i pulsira ga. Trial and error sa podesavanjem otkucaja, baze i amplitude. Na kraju su ovi brojevi ispali pristojni:

```
const otkucaj = 0.01; // manji brojevi, brze kuca
const baza = 3;      // baza ("Minimalno" srce)
const amplituda = 1.2; // amp (sinusna faza, "maksimalno" srce)
```

```
const faktor = Math.pow(Math.sin(t * otkucaj * Math.PI), 2);
const skala = baza + amplituda * faktor;
```

`pi` - ispisuje decimale PI jednu po jednu.

`ctx.fillText(text.slice(0, i1), 10, canvas.height / 2);` // **klasican .slice. ako je slice indeks manj od duzine teksta, povecaj indeks, pozovi writePi posle 200 ms i u krug.**

```
    if (i1 < text.length) {
        i1++;
        setTimeout(writePi, 200);
    }
}
```

7.1.11. Definisanje kontejnera

```
const container = document.getElementById('glossary_container');
const modal = document.getElementById('modal_container');
```

7.1.12. kartica za svakog

```
topics.forEach(topic => {
  const card = document.createElement('div');
  card.className = 'glossary_card';
```

```
  const img = document.createElement('img');
  img.src = topic.img;
  img.alt = topic.title;
  card.appendChild(img);
```

```
  const h3 = document.createElement('h3');
  h3.textContent = topic.title;
  card.appendChild(h3);
```

```
  const p = document.createElement('p');
  p.textContent = topic.description;
  card.appendChild(p);
```

7.1.13. otvaranje modala

```
card.addEventListener('click', () => {
  const paragraphs = topic.moreInfo.map(p => `

${p}

`).join("");
  modal.innerHTML = `
    <div class="modal_overlay"></div>
    <div class="modal_content" style="background-image: url('${topic.bgImg}');">
      <span class="modal_close">&times;</span>
      <div class="modal_text_overlay">
        <h2>${topic.title}</h2>
        ${paragraphs}
      </div>
      <canvas class="modal_diagram"></canvas>
    </div>
  `;
  modal.classList.add('show');
```

7.1.14. Zatvaranje modala

```
modal.querySelector('.modal_close').addEventListener('click', () =>
modal.classList.remove('show'));
modal.querySelector('.modal_overlay').addEventListener('click', () =>
modal.classList.remove('show'));
```

7.1.15. Priprema tačaka za dijagram i poziv

```
const diagramPoints = [
  { x: 0.05, y: 0.8 }, { x: 0.25, y: 0.4 }, { x: 0.5, y: 0.6 }, { x: 0.75, y: 0.3 }, { x: 0.95, y:
0.5 }
];

drawDiagram('.modal_diagram', topic.diagramType);
```

```
});
```

7.1.16. Ubacivanje kartice

```
    container.appendChild(card);  
  });
```

```
    const navToggle = document.querySelector('.nav_toggle');  
    const navList = document.querySelector('.nav_list');
```

Otvori/zatvori burger

```
    navToggle.addEventListener('click', () => {  
      navList.classList.toggle('show');  
    });  
  });
```