



SAPIENTIA
ERDÉLYI MAGYAR
TUDOMÁNYEGYETEM

Szoftver rendszerek tervezése

~My life tracker~

Veztőtanár: Dr. Szántó Zoltán

Készítette: Szabó Huba

Tartalom

Bevezető.....	2
Projekt célja.....	2
Követelmény specifikáció.....	2
• Felhasználói követelmények.....	3
• Use case diagramm.....	4
• Funkcionális rendszerkövetelmények.....	3
• Nem funkcionális rendszerkövetelmények.....	3
Tervezés.....	6
Összegzés.....	15

Bevezető

Mivel a mai világban egyre előtérbe helyeződik az online térben való manifesztáció, ezért egy weboldal simán helyettesíteni tudja egy személyi névjegykártya a szerepét.

Mivel az egyetemi éveim során végigjárt kurzusaimat, csak jómagam tudom, ezért arra gondoltam, hogy remek ötlet lenne az, hogy egy olyan online platformot készítek magamnak, ami segítségével könnyen megtudhatják az ismerőseim, illetve a jövődöbéli esetleges alkalmazóim, hogy milyen képességeim vannak.

A projekt célja

Munkánk során egy olyan weboldalt szerettem volna megvalósítani, amely segítségével egy jól informált személyes képet tud alkotni a felhasználó rólam.

Nagyon fontos cél volt a letisztultság valamint az, hogy aki használja az oldalt, könnyen tudjon kapcsolatot fenntartani velem, és ha bármivel kérdése van, a legyorsabb úton eljusson az üzenete hozzám.

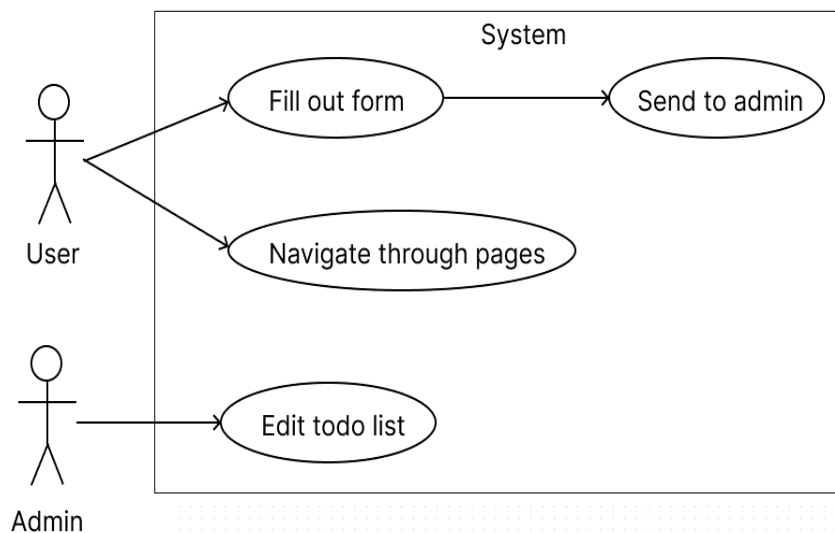
Követelmény specifikáció

Felhasználói követelmények

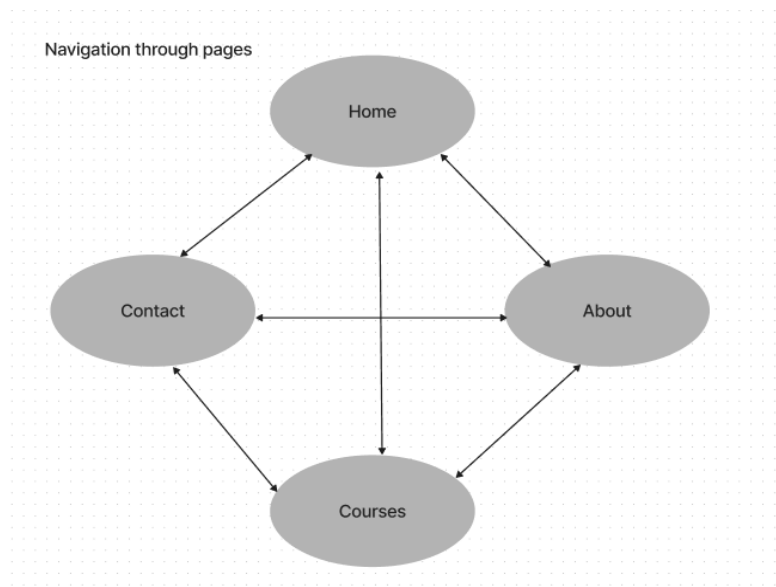
- A felhasználó amint megnyitja a weboldalt a főoldalt(home paget) látja elsőként. A „Follow me” gombbal átnavigálhat a egy olyan oldalra ahol az adminról kaphat információt.
- A főoldalon lefele görgetve, információkat szerez valamint gombokra való kattintással elkerülhet a weboldal más oldalaira.
- A navigációt a felhasználó számára a weboldalon jelenlévő navigációs menü segíti.
- Fontos, hogy a felhasználó tudjon üzenetet küldeni az adminnak ezért erre külön kontak oldal áll rendelkezésére.

- Az információ legyen jól megfogalmazva, értelmesen kivitelezve a felhasználó számára.
- A navigáció érthető legyen, ne legyen túl komplex
- Fontos, hogy a felhasználó az üzenetküldéshez a következő adatait kell megadja:
 - Név
 - Telefonszám
 - E-mail
 - A beszélgetés témája(subject)

Use case diagram



1. ábra



2. ábra

Rendszerkövetelmények

Funkcionális rendszerkövetelmények

- Amint megnyitja a weboldalt a főoldalt látja elsőként. Itt egy általános bemutatás lesz a kurzusaimról, az egyetemi kampusról, tevékenységekről. Mivel a főoldal szekciókból van felépítve, minden fentebb felsorolt témakörnek van egy saját szekciója.
- A webfiók oldalain a bal, felső sarokba fog lenni egy SAPIENTIA logo, ami az egyetemet képviseli. Erre kattintva, mindig visszaviszi a felhasználót a „HOME” oldalra.
- A főoldalon a felhasználó látni fog egy gombot, ezzel a szöveggel, hogy „Follow me!”, amire ha rákattint az „ABOUT” oldalra fog vezényelni.
- A főoldalon látható egy olyan szekció, amiben egy gombra kattintva („Contact me”) a „CONTACT” oldalra fog dobni, ahol üzenetet hagyhatunk a weboldal tulajdonosának(adminnak).
- A internetes fiók minden oldalán, jobb felső sarokban fog elhelyezkedni egy navigációs menü: HOME, ABOUT, COURSES, CONTACT oldalakkal.
- Az „ABOUT” fülre kattintva, az oldalon rolam fog információt találni, melyek engem, mint weboldal tulajdonost írnak le.
- A „COURSES” fülre kattintva, az általam teljesített kurzusok listáját tartalmazó oldalra fog navigálni.
- A „CONTACT” fülre kattintva, egy olyan oldalra jutunk, ahol a megfelelő információk megadásával, üzenetet tudunk küldeni(adminnak).
- Minden oldal alján szerepelek gombocskák, melyekre kattintva a különböző online platformokon lévő admin fiókhoz juthatunk el(Linked-in, Github, Instagram, Csillag).

Nem-funkcionális rendszerkövetelmények

A weboldal megfelelő működéséhez a következő rendszerkövetelményeknek kell eleget tenni:

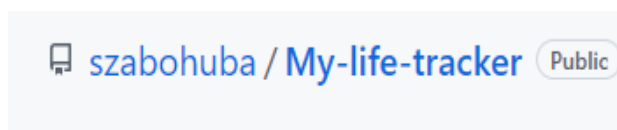
- Laptop(egér)/PC/Okostelefon
- Internet hozzáférés
- Kijelző(a design maximális kivitelezése miatt nagyobb felbontás ajánlott)
- Böngésző(Google Chrome/Brave/Opera/Safari/Microsoft Edge)
- Legalább 4 giga ram a optimális használathoz(Böngésző optimális működése)
- Kommunikáció esetén online levelező fiók

Tervezés

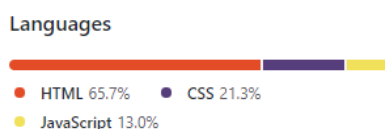
A tervezés folyamat elején eldönt, hogy a weboldalamnak milyen oldalai lesznek és hogyan lehet közöttük navigálni.

Hogyha már az alap oldalak ötletei megvoltak, következő fázisban elhatároztam, hogy az eredetiség érdekében html, illetve css nyelveken valósítom meg a weboldal struktúráját illetve css segítségével fogom ezt dizájnolni. A háttérben történő műveleteket pedig JavaScript segítségével oldom meg.

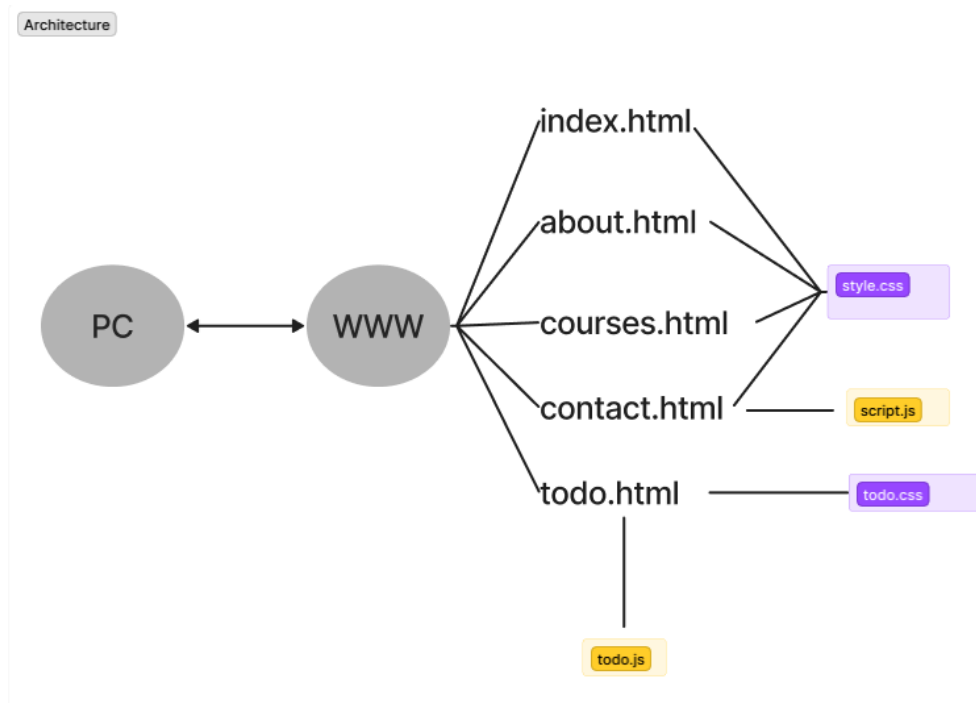
Az így létrehozott projektet a My-life-tracker GitHub repository-hoz csatoltam. Ezek után a projekt fejlesztésének lépéseit szintén Github Knaban board segítségével követtem, amit minden jelentősebb fejlesztés után frissítettem.



3. ábra



4. ábra



5. ábra

Első lépésben, megterveztem egy minimális fájlrendszer, ami köré egy hatékony navigációt tudok építeni. Elsőnek a Home oldallal foglalkoztam. Itt szekciókra struktúráltam az oldal tartalmát.

Miután a főképernyő hátterét beállítottam, a weboldal egyik legfontosabb komponensével, a navigációs menüvel foglalkoztam, ami tulajdonképpen egy lista, ami, horizontálisan van elhelyezve, illetve dizájnolva(5, 6. ábrák).



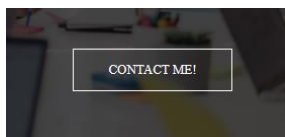
6. ábra

```
<nav>
  <a href="index.html">
    
  </a>

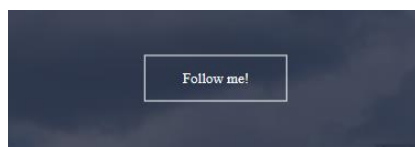
  <div class="nav-links" id="navLinks">
    <i class="fa fa-times" onclick="hideMenu()"></i>
    <ul>
      <li><a href="index.html">HOME</a></li>
      <li><a href="about.html">ABOUT</a></li>
      <li><a href="courses.html">COURSES</a></li>
      <li><a href="contact.html">CONTACT</a></li>
    </ul>
  </div>
  <i class="fa fa-bars" onclick="showMenu()"></i>
</nav>
```

7. ábra

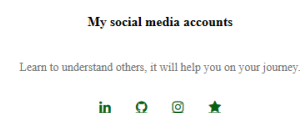
A navigáció szempontjából, nagyon fontos szerepet játszanak a gombok, melyek ha egy szekcióban szereplő információhoz navigálnak(8, 9, 10. ábrák).



10. ábra



9. ábra



8. ábra

A CONTACT ME gomb azon oldalra navigál ahol a felhasználó üzenetet tud hagyni nekem, míg a FOLLOW ME akciógomb az ABOUT oldalra navigál, ahol rólam van leírás. A navigáció, nem csak oldalon belültre terveztem, hanem megadtam a Github, Gmail, Instagram elérhetőségeimet is(10. ábra). Ezt a footer részbe helyeztem, hogy minden oldalon megjelenjenek, a könnyű elérés érdekében(11. ábra). A gombok, illetve sima szöveg helyett, itt ikonokat használtam a esztétikusabb design érdekében. Ezeket a font awesome könyvtár segítségével helyeztem be a projektbe.

```
<section class="footer">
  <h4>My social media accounts</h4>
  <p>Learn to understand others, it will help you on your journey.</p>

  <div class="icons">
    <a href="https://www.linkedin.com/in/huba-szabo-0b47261bb/" class="fa fa-linkedin"></a>
    <a href="https://github.com/szabohuba" class="fa fa-github"></a>
    <a href="https://www.instagram.com/tailor_hubi/" class="fa fa-instagram"></a>
    <a href="todo.html" class="fa fa-star"></a>
  </div>
</section>
```

11. ábra

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@6.2.1/css/fontawesome.min.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

12. ábra

A főoldalnak egy nagyon fontos navigációs eszköze a logó, amit az egyetem logója képvisel. Erre kattintva mindig a főoldalra jutunk vissza.

```
<a href="index.html">
  
</a>
```

13. ábra

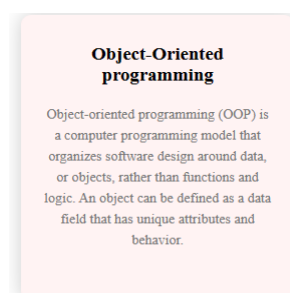


14. ábra

A főoldalon a továbbiakban, minden szekció ismerteti valamit az egyetemmel kapcsolatban. Az első szekcióban három tantárgy van kiemelve, ezek nevük van hangsúlyozva, valamint röviden ismertetve. A programozási szempontból ezen az oldalon, a design volt a cél, hogy egy letisztult, minimalista kinézetet adjak az oldalnak. Minden tárgy egy box designot kapott egy árnyék effektussal, ami akkor jelenik meg ha az egeret rávisszük (16. ábra).

```
.course-col:hover{
  box-shadow: 0 0 20px 0px rgba(0,0,0,0.2) ;
}
```

15. ábra



16. ábra

A következő szekcióban a Sapientia egyetem kampuszait jelenítettem meg. Programozási szempontból itt az volt a cél hogy mikor az egyetem valamelyik kampuszára ráviszed az egeret, akkor Sapi zöld szín jelenjen meg és egy animáció segítségével, egy ideiglenes állapotban kiírja, hogy a képen látható kampusz, hol helyezkedik el(17. ábra).



17. ábra

A Facilities szekcióban pár egyetemi tevékenységet soroltam fel amiket az egyetem keretein belül lehet igénybe venni.

Az ez után következő szekcióban egy szövegboxot jelenítettem meg melyen a fentebb említett navigációs gomb található, ami a contact oldalra kalauzol.

Az about.html oldalon, magamról mint személyről, mint csapatjátékosról valamint, a kedvenc időtöltésemről írtam pár gondolatot, ugyanakkor ez színesítése okából pár képet is beraktam.

A courses.html oldalon hasonlóképpen, mint az ABOUT oldalon, megjelenítettem a kurzusaimat, melyek mellé egy leíró képet társítottam(18. ábra). Ezeket az oldalakat(about, courses) az egységes dizájn fogja össze. Mindkét oldalon megtalálható a navigációs menü vagy sáv, valamint a footer részlegben az elérések.

Software testing

Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs and improving performance.



18. ábra

Elérkeztünk a projekt legkomplexebb részéhez, a kommunikáció megvalósításához. Ez azt jelenti, hogy a contact.html oldalon, egy contact formot alakítottam ki, ahol a bizonyos adatokat beírva a bemeneti mezőkbe, a felhasználó, vagy a vendég képes üzenetet küldeni nekem email formában. A contact-form tartalmaz text típusú bemeneteket, telefon illetve mail bemeneteket(19. ábra). Ezek után egy akciógomb található, melyet ha lenyomjuk (onClick()), meghívódik a paraméternek adott JavaScript sendMail() nevű függvénye. Ez a függvény a script.js filban van implementálva.

Connect with me!

Send me your thoughts!

Name:

Phone:

Your name

Your phone number

E-mail:

Subject:

Your mail adress

Subject

Message:

Your message

Submit

Reach me!

Email:

huba.sz22@gmail.com

Phone:

+4072334422

Adress:

Str. XXXXXX, NR YYYYYYY, Mures, Mures

Post code:

515515

19. ábra

A sendMail() függvény egy nagyon egyszerű funkció, melyben paraméterként lekérjük a 19. ábrán látható adatokat, es ezeket a emailjs.send metódussal elküldjük a mailjs szervernek(20. ábra), ahol készítettem egy példa mail formátumot, amilyen formában szeretném kapni az értesítést(21. ábra).

```
function sendMail(){
  var params={
    name_id : document.getElementById("name_id").value,
    mail_id : document.getElementById("mail_id").value,
    phone_id : document.getElementById("phone_id").value,
    subject_id : document.getElementById("subject_id").value,
    message_id : document.getElementById("message_id").value
  }

  emailjs.send("service_y9r1zlx","template_ih8zbx",params).then(function(res){alert("Success, your mail has been registered! " + res.status);
});
}
```

Content Auto-Reply Attachments Contacts Settings

Subject *

New message from {{name_id}}

Content *

B I U

font-family: sans-serif; font-size: 12pt;

Hello,

You got a new message from {{name_id}}.

Email: {{mail_id}}

Telephone: {{phone_id}}

Subject: {{subject_id}}

Message: {{message_id}}

To Email *

redpapper11@gmail.com

From Name

Szabo Huba

From Email *

☒ Use Default Email Address

Reply To

{{reply_to}}

Bcc

Cc

20. ábra

21. ábra

Az `emil.Send()` funkcióban használt `emailjs.send()` függvény paraméterei megtalálhatóak a 22. ábrán.

```
emailjs.send(serviceID, templateID, templateParams, publicKey);
```

22. ábra

Parameters

NAME	TYPE	DESCRIPTION
<code>serviceID</code>	String	Service ID of the service through which the email should be sent. Reserved keyword default_service is supported, and should be used to use the default service, which can be set and changed via EmailJS dashboard.
<code>templateID</code>	String	Template ID of the email
<code>templateParams</code>	Object	Template parameters of the template.
<code>publicKey</code>	String	(optional) Public Key of the account. It's not required if you used the <code>init</code> method.

A `serviceID` paraméter segítségével küldjük a szervernek az adatot, hogy a második paraméternek megadott `templateID` szerinti formába alakítsa. A harmadik paraméternek adjuk meg a konkrét adatot, amit majd a template-be kerül. A `publicKey` pedig egy opcionális paraméter itt, de én a html filomba adtam hozzá ezt a paramétert a head részlegben, ahol a javascript file-t include-oltam (23, 24. ábra).

```
<script type="text/javascript"
  src="https://cdn.jsdelivr.net/npm/@emailjs/browser@3/dist/email.min.js">
</script>

<script type="text/javascript">
(function(){
  emailjs.init("CXHogjOhghho6iKv8");
})();

</script>
```

24. ábra

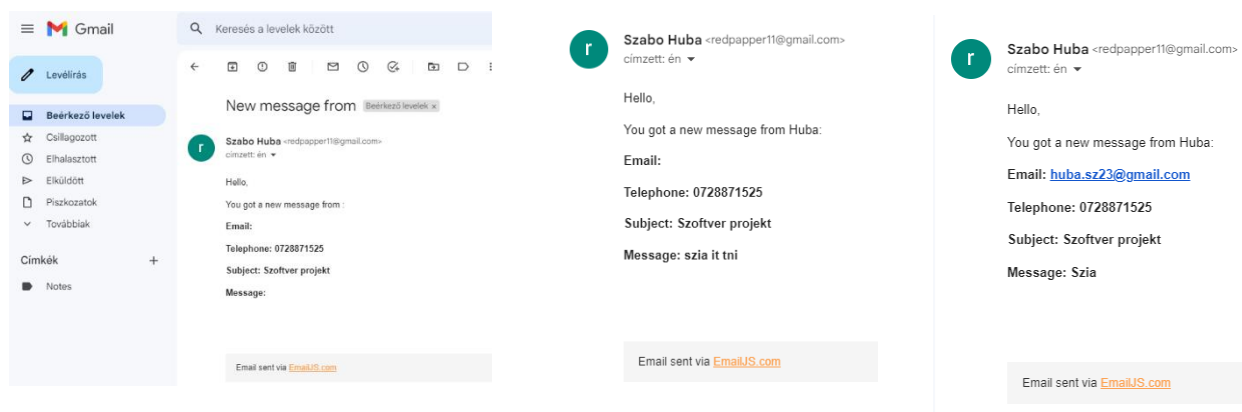
Public Key

CXHogjOhghho6iKv8

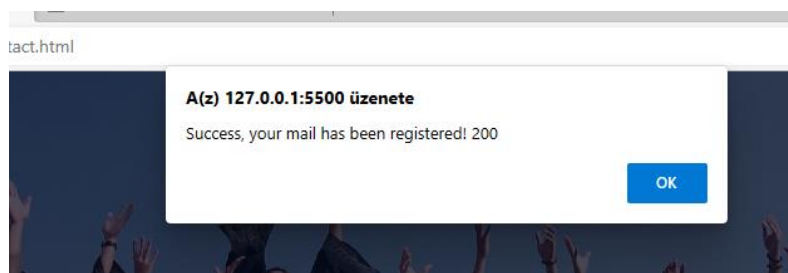
23. ábra

A 20. ábrán még megfigyelhető, hogy van egy alert is a send-en, hogy egy visszajelzése legyen a felhasználónak, hogy az üzenete el lett küldve(25. ábra).

A továbbiakban a levelezés tesztelésével foglalkoztam. Megbizonyosodtam róla, hogy több esetben is működik a szerver, valamint a hibáimat javítottam(24. ábra).



24. ábra



25. ábra

A projekt tartalmaz egy todo.html, egy todo.js, egy todo.css file-t is. Ez egy template kanbanboard, amit egy okos opciónak tartottam betenni a projektembe. A todo.css file-t azt magam dizájnoltam, a weboldalamnak a színeire.

Ezt az oldalt a footerben levő csillag ikonra kattintva érhetjük el, és a 28. ábrán lévő oldalra navigál.

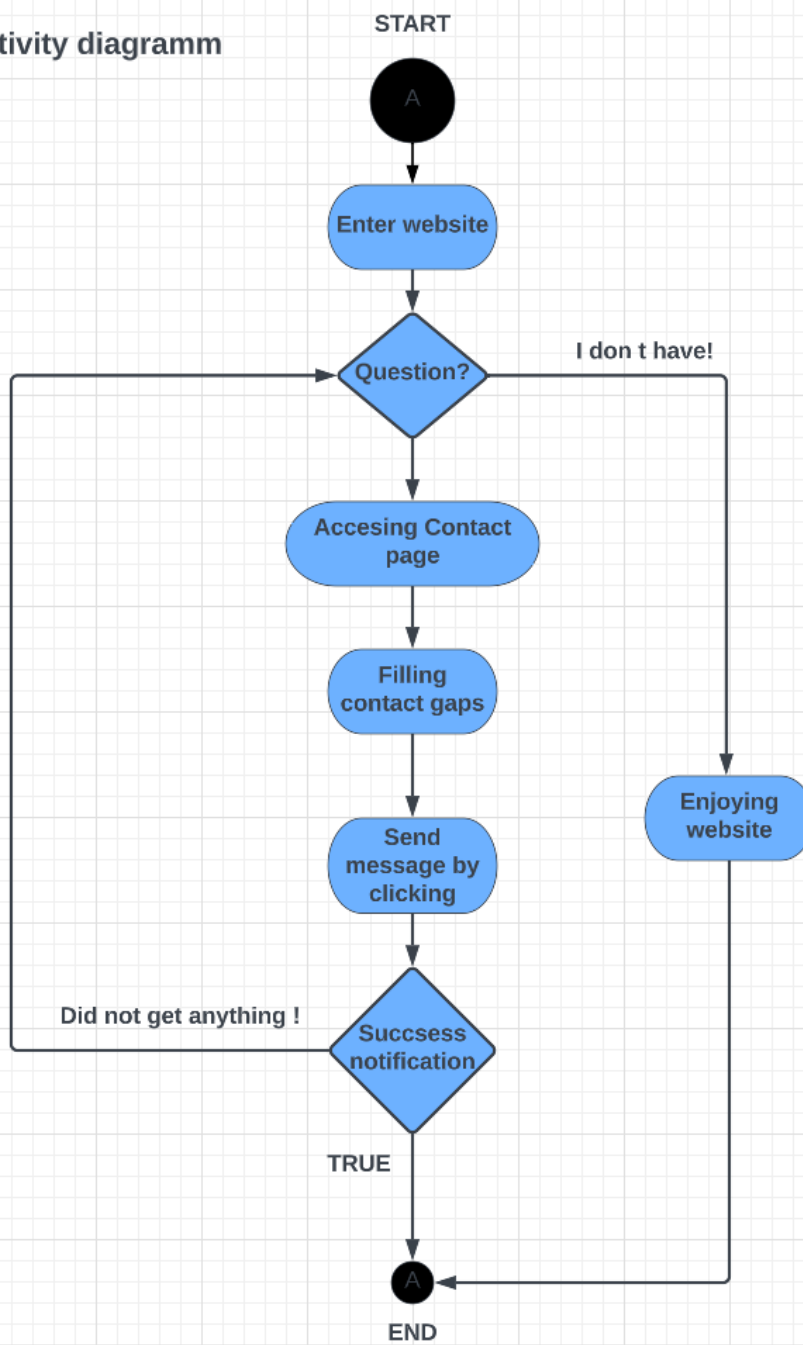
My social media accounts

Learn to understand others, it will help you on your journey.

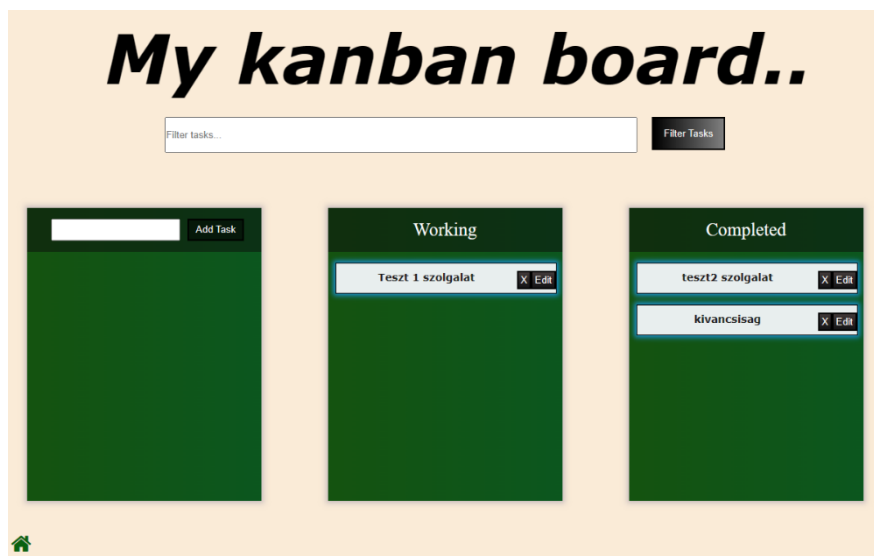


26. ábra

Activity diagramm

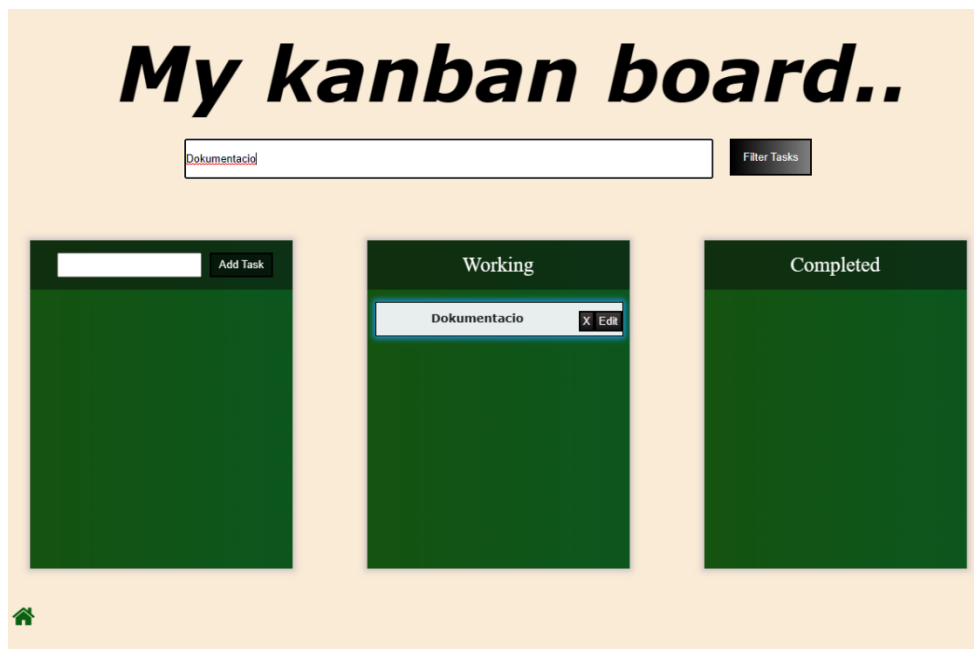


26 .ábra



27. ábra

Ez a tevékenysékövetőnek három oszlopa van. Az elsőben hozzátudunk adni egy tervben lévő tennivalót. Ezt követően ha nekifogunk teendőknek, akkor áttudjuk húzni a working dobozba, ami azt jelenti, hogy az művelet folyamat alatt van. Az jobboldali dobozba pedig azokat rakjuk, mint a neve is mutatja, amik megvannak. Ez a kis funkció, local storage alapon működik, ami konkrétan azt jelenti, hogy minimális adatot tudunk elmenteni a böngészőben. A keresési sáv segítségével tudunk keresni a tevékenységink között, eltüntetni a többit és megjeleníteni a keresett todo-t(28. ábra). A bal alsó sarokban, pedig a ház ikonra kattintva vissza tudunk navigálni a főoldalunkra.



28. ábra

Összegzés

Továbbfejleszthetőségek

A projektem rengeteg képpen továbbfejleszthető. Az első és legfontosabb egy adatbázis létrehozása, melynek az lenne a feladata, hogy komplexebb tevékenységeket tudjunk elmenteni a tevékenységkövetőnkől.

Másik továbbfejleszthetőségi opció, a kurzusok tárolása szintén a fentebb említett adatbázisba.

Mivel az szoftverfejlesztés egy globális kapcsolatokkal rendelkező munkakör, nagyon hasznos lenne szerintem az oldal megejelítése további nyelveken, példáulnaok okáért német, angol.

