

UNIVERSITATEA “ALEXANDRU IOAN CUZA” DIN IAȘI

FACULTATEA DE INFORMATICĂ



LUCRARE DE LICENȚĂ

FII Dashboard

propusă de

Teodora Chiosa

Sesiunea: Iulie, 2018

Coordonator științific

Drd. Colab. Florin Olariu

UNIVERSITATEA “ALEXANDRU IOAN CUZA” DIN IAȘI

FACULTATEA DE INFORMATICĂ

FII Dashboard

Teodora Chiosa

Sesiunea: Iulie, 2018

Drd. Colab. Florin Olariu

DECLARAȚIE DE CONSIMȚĂMÂNT

Prin prezenta declar că sunt de acord ca Lucrarea de licență cu titlul „FII Dashboard”, codul sursă al programelor și celelalte conținuturi (grafice, multimedia, date de test etc.) care însoțesc această lucrare să fie utilizate în cadrul Facultății de Informatică.

De asemenea, sunt de acord ca Facultatea de Informatică de la Universitatea „Alexandru Ioan Cuza” din Iași, să utilizeze, modifice, reproducă și să distribuie în scopuri necomerciale programele-calculator, format executabil și sursă, realizate de mine în cadrul prezentei lucrări de licență.

Iași, data _____

Absolvent *Teodora Chiosa*

(semnătura în original)

CUPRINS

Aspecte teoretice.....	1
User Experience	1
Ce este și de ce este important?.....	1
UX Personas.....	3
UX Writing.....	5
Principii fundamentale ale interacțiunii	6
Care e diferența între UX și UI?	7
User Interface.....	8
Alegerea elementelor de interfață	8
<i>Best practices</i> pentru un design de interfață	8
Procesul UI Design	9
UI Development.....	11
SCSS	11
SCSS - Avantaje.....	12
SCSS - Dezavantaje	14
CSS3 – noutăți și obstacole.....	14
Front-end.....	15
Aspecte practice.....	16
User experience.....	16
User interface	31
UI development.....	35
Posibile îmbunătățiri.....	39
Concluzii.....	41

Introducere

Prin lucrarea de față doresc să vin în ajutorul studenților Facultății de Informatică și să ofer informații centralizate asupra notelor, taxelor, prezențelor, orarului, materialelor de studiu, temelor, noutăților din cadrul facultății, prin intermediul unei interfețe prietenoase.

În momentul actual, informația este dispersată. Lipsa de uniformitate și lipsa update-urilor aduse site-urilor, atât ca interfață cât și ca tehnologii utilizate, nu prezintă un beneficiu pentru imaginea Facultății de Informatică și nu reflectă ideea de inovație în domeniu.

Lipsa de uniformitate se exemplifică prin următoarele:

- folosirea multiplelor conturi pentru accesul la informații:
 - o pe eSIMS se pot găsi datele personale ale studentului, notele finale și obligațiile de plată, autentificarea fiind făcută pe baza numărului matricol
 - o pe emailul facultății (Webmail) se primesc notificări de la profesori sau secretare cu privire la notele de pe parcursul semestrului, adeverințele CTP, orarul obișnuit și cel de examene
 - o fără nici un cont, oricine are acces la orar și notele intermediare sau finale ale studenților ce se află pe paginile publice ale profesorilor (cu excepția celor protejate de o parolă)
- pentru a avea acces la materialele și notele unor discipline, studentul trebuie să rețină numele site-ului profesorului și, unde este cazul, parolele anunțate la curs/laborator
- formula de calcul a notei finale diferă de la o disciplină la alta și trebuie căutată în fișa disciplinei
- se observă lipsa de update-uri, starea nefinalizată a paginilor, ceea ce duce la o experiență nu tocmai user-friendly pe aceste site-uri; de exemplu: titlul paginii eSIMS este “Untitled page”, limbajul folosit pe prima pagină nu inspiră profesionalism, butonul de autentificare este greu de reperat la o primă accesare a paginii
- inconsistența este relevantă și de paginile profesorilor: întâlnim site-uri ușor de folosit dar și site-uri *plain text* ce nu folosesc link-uri, navigarea fiind făcută prin a tasta un nume de pagină direct în URL

Aceste aspecte au determinat studenții să comunice pe diverse canale (în principal Facebook) pentru a face rost de materiale de studiu, parole, e-mail-uri etc, aceleași întrebări fiind repetate în fiecare semestru.

Având în vedere acestea, voi realiza prin prezenta lucrare:

- crearea unui portal special destinat studenților
- folosirea unui singur cont pentru toate informațiile prezentate mai sus
- restructurarea și rearanjarea informațiilor în pagină pentru a facilita experiența utilizatorului pe acest site
- standardizarea paginilor profesorilor pentru a face mai ușoară găsirea informațiilor
- posibilitatea înregistrării prezențelor online
- includerea știrilor site-ului Facultății de Informatică
- includerea notificărilor legate de Webmail
- aducerea unei perspective fresh, în concordanță cu trendurile actuale din domeniile UI/UX și Front-end

Primul capitol al acestei lucrări, intitulat „Contribuții”, va prezenta elementele inedite pe care le aduce această lucrare, și anume unificarea site-urilor existente în prezent, sub accesul unui singur cont, cu o interfață ușor de folosit/user-friendly.

În paginile ce urmează, voi trata următoarele aspecte în această ordine:

- UX: experiența utilizatorului pe acest site (e.g. cât de intuitivă și rapidă este navigarea în aplicație, găsirea de informații fără multe confuzii și click-uri),
- UI: design-ul acestuia (e.g. cât de plăcut din punct de vedere estetic este, dacă reflectă imaginea universității/facultății, dacă inspiră profesionalism, modernizare și inovație)
- Preprocesoare CSS (mai exact SASS/SCSS): avantaje, reutilizarea codului/DRY, folosirea optimă, arhitectura stilurilor
- CSS 3: noutăți
- Framework-ul Angular 5: folosirea acestuia împreună cu SASS și HTML

Tehnologiile/plugin-urile folosite vor fi: Angular 5, SASS/SCSS, componente Angular Bootstrap (fără CSS).

Codul sursă al aplicației se poate găsi pe Github la adresa:

<https://github.com/elenachiosa/FIIDashboard> iar un demo al acesteia se găsește pe Heroku (PaaS – *platform as a service* ce permite construirea și rularea aplicațiilor pe cloud):
<http://fiidashboard.herokuapp.com/>.

Contribuții

Având drept ghid funcționalitățile actual implementate de către facultate/universitate, am creat o aplicație web ce este mai ușor și mai plăcut de folosit de către studenți, oferind o înfățișare nouă și folosind tehnologii moderne.

Drept îmbunătățiri majore precizez:

- Experiența mai bună a utilizatorului pe site
- Design simplu și atractiv
- Posibilitatea de a vedea aplicația pe un ecran de mobil (responsive), un „must” al zilelor noastre
- Stilizarea aplicației folosind preprocesorul de CSS: SASS – un mare ajutor în etapele de dezvoltare și mentenanță a aplicației
- Folosirea limbajului Typescript pentru organizarea mai bună a codului și definirea mai strictă a variabilelor
- Separarea Front-end-ului de Back-end pentru modularitate

Aspecte teoretice

În cele ce urmează, voi detalia aspecte teoretice (folosind articole online sau imprimate și imagini) și practice (cu exemple din aplicație sau exemple externe).

După cum bine știm, Facultatea de Informatică ne oferă bazele programării și dorește să ne trezească interesul spre a învăța, nu numai ce se preda la cursuri, ci mai ales concepte, limbaje și tehnologii noi, capacitatea de adaptare la nou fiind esențială în domeniu.

Ieșit din băncile facultății, un student se va lovi de asemenea provocări, de colaborarea în echipă, de metode de lucru etc. Astfel își va da seama că scopul unei aplicații nu este doar să efectueze sarcinile detaliate de client, ci să ofere funcționalități într-un mod intuitiv, atractiv, ușor de folosit și eficient din punct de vedere al timpului.

User Experience

Ce este și de ce este important?

Potrivit definiției oferită de ISO (International Organization for Standardization), alineatul 3.23, Experiența utilizatorului este “percepția și reacția unei persoane, ce rezultă din folosirea și/sau folosirea anticipată a unui produs, sistem sau serviciu”, nota nr. 2 precizând și că este „o consecință a imaginii brand-ului, prezentării, funcționalității, performanței sistemului, comportamentului interactiv, capabilităților de asistență a sistemului interactiv, a stării interne și fizice a utilizatorului rezultate din experiențe precedente, atitudine, abilități, personalitate și context de utilizare”.

(ISO/IEC JTC 1/SC 7, 2016)

Într-o formă mai ușor de înțeles, UX este tot ceea ce gândești și simți în timpul folosirii unui produs, serviciu etc. Experiența poate fi una atât de plăcută încât este ireproșabilă, sau, în extrema cealaltă, atât de nesatisfăcătoare încât stârnește confuzie, nervozitate, regret asupra timpului pierdut și dorința puternică de a renunța la folosirea acelui produs.

Un rol foarte important în UX îl are experiența anterioară a utilizatorului în contextul produselor cu rol similar: spre exemplu, pentru închiderea unui element (ce poate fi o fereastră de sistem de operare, un link spre *Sign in/Sign out*, sau un modal într-o aplicație web, utilizatorul și-a format reflexul de a îndrepta cursorul mouse-ului spre colțul din dreapta

sus al ecranului). Rolul unui UX designer este de a intra în pielea unui utilizator obișnuit și de a empatiza cu nevoile și obiceiurile sale, în același timp urmând trendurile actuale în domeniu, pentru crearea unei interfețe adecvate audienței țintă.



Figura 1 - User Experience Honeycomb, Peter Morville (Morville, 2004)

Figura 1 - User Experience Honeycomb, Peter Morville dorește să ilustreze principalele atribute specifice UX: valoros, ușor de găsit, folositor, utilizabil, dezirabil, accesibil și credibil.

În procesul său de lucru, un UX designer va trebui să își pună următoarele întrebări referitoare la produsul ce urmează a fi dezvoltat:

Produsul poate fi folosit? (Usable)

Dacă produsul final prezentat în fața clientului este incomplet sau prost făcut și deci nu poate fi folosit, continuarea interacțiunii cu acesta ar fi în van. Acesta este atributul de bază.

Poți găsi ușor informația de care ai nevoie? (Findable)

Este meniul intuitiv? Bara de căutare este unde te-ai aștepta să fie? Dacă trebuie să depui efort pentru a găsi ceva, înseamnă că UX-ul lasă de dorit. Același lucru îl putem spune și despre drumul parcurs până la produsul precizat: e.g. motorul de căutare a găsit ce trebuie în primele rezultate, sau nu s-a investit în Metadata pentru indexarea paginilor aplicației?

Produsul satisface nevoile clientului? (Useful)

Există situații în care designul și facilitatea de operare sunt impecabile, însă produsul nu oferă ceea ce clientul dorește iar acesta își va pierde rapid interesul.

Dorești să folosești produsul? (Desirable)

Designul este intuitiv? Este o adevărată plăcere să folosești acel produs? Atunci vei dori să-l folosești din nou, poate chiar să îl recomanzi altora.

Produsul are valoare? (Valuable)

Poți spune că ai salvat timp, efort fizic, bani sau că nu ai mai văzut ceva similar pe piață? Dacă un produs nu are valoare, nu va prezenta interes pentru mult timp, oricât de frumos și ușor de folosit ar fi.

Ai încredere în produs? (Credible)

Se aplică mai ales magazinelor online (nu ai dori să dezvălui datele cardului de credit oricui). Limbajul folosit în scrierea conținutului, designul atractiv și minimalist pot contribui la o imagine profesională și credibilă.

Este produsul accesibil? (Accesible)

Oricine ar trebui să aibă acces la produs, indiferent de dizabilitățile sale.

(DeRome, 2015)

UX Personas

O *persona* reprezintă un grup de utilizatori ce manifestă tipare comportamentale similare precum: decizii de cumpărare, utilizarea de tehnologii sau produse, stilul de viață etc.

(O'Connor, 2011)

De ce avem nevoie de *personas*?

- Pentru a ști care este audiența țintă a produsului
- Pentru a înțelege mai bine care sunt problemele și nevoile audienței

- Pentru a concentra efortul echipei asupra creării unui produs utilizabil
- Pentru a nu întâmpina discuții legate de funcționalități ce nu vor fi utilizate
- Pentru ca echipa de dezvoltatori să înțeleagă mai bine motivele și modalitățile de interacțiune cu produsul

Un număr ideal de *personas* ar fi până la 4, ideea fiind explorarea și adresarea nevoilor diverșilor utilizatori în vederea creării unui produs pentru o audiență cât mai largă, dar fără a considera target pe toată lumea. Spre exemplu, pentru un produs destinat unei companii mari, cu utilizatori care poartă diverse titluri și au pregătiri diferite (șefi, muncitori obișnuiți, etc.), va fi nevoie de un număr mai mare de *personas* pentru a trata toate aceste categorii de angajați.

(Tools, 2014)


Persona:	USDA Senior Manager Gatekeeper	Goals and tasks:	He is focused, goal-oriented within a strong leadership role. One of his concerns is maintaining quality across all output of programs.
Photo:			Spends his work time: <ul style="list-style-type: none"> • Requesting and reviewing research reports, • preparing memos and briefs for agency heads, and • supervising staff efforts in food safety and inspection.
Fictional name:	Matthew Johnson	Environment:	He is comfortable using a computer and refers to himself as an intermediate Internet user. He is connected via a T1 connection at work and dial-up at home. He uses email extensively and uses the web about 1.5 hours during his work day.
Job title/ major responsibilities:	Program Staff Director, USDA		
Demographics:	<ul style="list-style-type: none"> • 51 years old • Married • Father of three children • Grandfather of one child • Has a Ph.D. in Agricultural Economics. 	Quote:	"Can you get me that staff analysis by Tuesday?"

Figura 2 - Persona dezvoltată de către Departamentul U.S. al Agriculturii (USDA) Economic Research Service (ERS)

După cum se vede în Figura 2 - Persona dezvoltată de către Departamentul U.S. al Agriculturii (USDA) Economic Research Service (ERS), elementele unei *persona* sunt următoarele:

- grupul din care face parte (e.g. manager)
- numele fictiv

- funcția la locul de muncă și responsabilitățile sale
- vârsta, educația, etnia, genul, statutul familial
- motivul pentru care ar folosi produsul dezvoltat
- starea fizică, socială și tehnică
- un citat ce rezumă funcționalitatea dorită de la produs
- o poză ce reprezintă această *persona*

(Usability.gov, Personas, fără an)

UX Writing

Un UX writer folosește cuvinte pentru a crea interfețe, prin a scrie conținut text (numit „copy”) ce ajută utilizatorul să-și atingă scopul, să înțeleagă mai bine ce urmări au acțiunile sale într-o aplicație. Pentru a crea interfețe mai ușor de utilizat, conținutul nu trebuie lăsat la sfârșit. Dacă ne gândim la site-ul nostru ca la o conversație cu utilizatorul, conținutul are prioritate, și nu amplasarea sa în pagină.

Pe scurt, UX writing înseamnă:

- Scrierea de conținut pentru design, unde cuvintele au scop și au în vedere dorințele utilizatorului (Figura 3 - exemplu de UX writing pe Mailchimp - oferirea de sugestii relevante în procesul de logare)
- Luarea în considerare a modului de gândire și a sentimentelor utilizatorului
- Evitarea textului robotic și impersonal

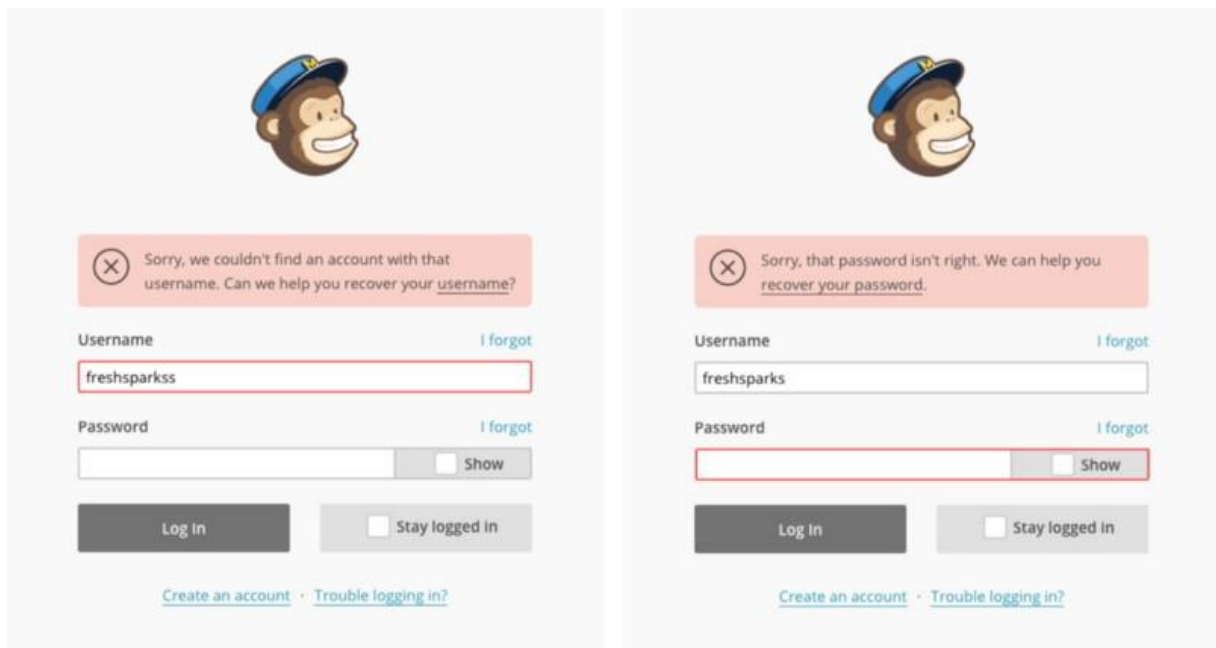


Figura 3 - exemplu de UX writing pe Mailchimp - oferirea de sugestii relevante în procesul de logare

(uxwriter.cc, 2017)

Principii fundamentale ale interacțiunii

Experiența cu un produs este crucială deoarece determină cu cât drag își amintește utilizatorul de interacțiunile sale cu acel produs. Experiența în ansamblu a fost una pozitivă sau frustrantă și confuză? Când interacționăm cu un produs, trebuie să ne dăm seama cum funcționează (ce face, ce operațiuni sunt posibile), ceea ce vom numi detectabilitate (discoverability). Câteva concepte ce o alcătuiesc sunt următoarele: *affordances*, *signifiers* și *feedback*.

Affordances

Termenul de *affordance* se referă la o relație între proprietățile unui obiect și capacitățile agentului ce determină cum poate fi folosit acel obiect. Spre exemplu, sticla permite transparență dar în același timp blochează trecerea altor obiecte fizice – deci sticla ne permite suport și posibilitatea de a vedea prin ea dar nu și trecerea aerului sau a altor obiecte (cea din urmă fiind considerată *anti-affordance*, împiedicarea interacțiunii). Pentru a fi eficace, acestea două (*affordance*, *anti-affordance*) trebuie să fie sesizabile. Dacă nu pot fi percepute, atunci dorim semnalarea prezenței acestora prin *signifiers*.

Signifiers

În timp ce *affordance* determină ce acțiuni sunt posibile, *signifier* comunică unde trebuie să aibă loc acțiunea. Utilizatorii vor căuta indicii sau semne pentru a înțelege modul de folosire al produsului. Astfel, termenul de *signifier* se referă la orice semn sau sunet, orice indicator perceptibil ce comunică un comportament unei persoane (cuvinte, o ilustrație grafică, etc.).

Feedback

Feedback înseamnă comunicarea rezultatelor unor acțiuni. Acesta trebuie să fie instant (o întârziere de o secundă deja va ridica semne de întrebare iar dacă întârzierea este prea lungă utilizatorii vor renunța la folosirea unui produs). Nu în ultimul rând, *feedback*-ul trebuie să fie informativ (e.g. simpla atenționare asupra unei erori de sistem nu comunică suficient unui utilizator, pe când detalierea erorii și oferirea unor sugestii de acțiuni ajută enorm).

(Norman, 2013)

Care e diferența între UX și UI?

Adeseori, aceste două concepte sunt privite ca unul și același lucru. Deși în majoritatea cazurilor, UI designer-ul are cunoștințe de UX (având titlul de UI/UX designer), acestea sunt de fapt lucruri diferite. Ca o analogie, gândiți-vă la corpul uman: “UX reprezintă tot ce este intern și face corpul să funcționeze bine, iar UI este ce se află la suprafață, tot ce face corpul să arate frumos și atrăgător”.

(DeRome, 2015)

User Interface

Interfața utilizatorului se focusează pe anticiparea nevoilor utilizatorilor și asigurarea unei interfețe cu elemente ușor de accesat și ușor de înțeles. UI combină conceptele UX, Arhitectura Informației și Design.

Alegerea elementelor de interfață

Utilizatorii s-au familiarizat cu un anumit comportament al unor elemente de interfață; de aceea este indicat ca produsul dezvoltat să fie consecvent și predictibil în decizii și layout/așezarea elementelor. Astfel facilităm completarea eficientă a taskurilor utilizatorului și satisfacția acestuia.

Câteva elemente de interfață sunt:

- Input-uri – butoane, câmpuri text/numerice/etc., casete de bifare/selectare, dropdowns, toggles
- Componente de navigare – breadcrumbs, câmp de căutare, paginare, iconițe
- Componente de informare – tooltips, iconițe, bare de progres, notificări, modale, toasters
- Elemente container – acordeoane, tab-uri

Best practices pentru un design de interfață

Simplitatea – folosirea limbajului clar și a adnotărilor, evitarea elementelor inutile

Consecvența dată de elemente comune/generale – odată învățat comportamentul unui element, utilizatorul poate folosi acea informație în alte pagini sau zone ale aplicației

Aranjarea în pagină cu scop – Spațierea și ordinea elementelor în pagină dictează importanța acestora și sporește găsirea informațiilor relevante utilizatorului

Culorile și texturile – direcționarea atenției folosind culoare, luminozitate, contrast, textură (sau redirecționarea atenției de la un element la altul)

Tipografia în crearea ierarhiei și clarității – folosirea diferitelor mărimi, fonturi, aranjamente ale textului poate crește capacitatea de scanare și lizibilitatea

Comunicarea dintre sistem și utilizator – informarea în permanență relativ la acțiuni, schimbări în sistem, erori folosind elemente UI (modale, toasters, etc.)

Valorile predefinite – setările sau valorile default în câmpuri pot salva din timpul utilizatorului

(Usability.gov, User Interface Design Basics, fără an)

Procesul UI Design

Cunoscând elementele de bază cu care poate lucra și specificațiile clientului, un UI Designer poate începe să elaboreze interfața unei aplicații, creând o schiță, un wireframe, un mockup și în cele din urmă un prototip. Acestea sunt necesare pentru a clarifica împreună cu clientul care sunt așteptările sale asupra aplicației și pentru a salva timp și resurse financiare în comunicarea cu dezvoltatorii.

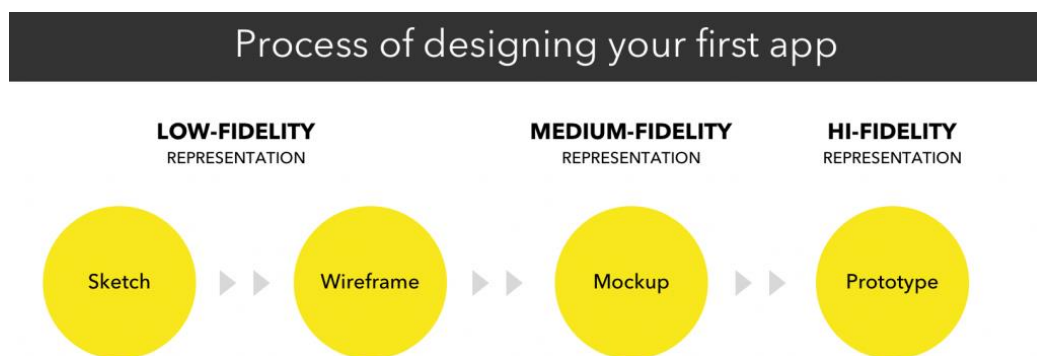


Figura 4 - Procesul de design al unei aplicații

Pe scurt, după cum se vede în Figura 4 - Procesul de design al unei aplicații, Figura 5 - Rezumat al procesului de UI design, se începe cu o schiță simplă ce poate fi realizată pe hârtie. Apoi wireframe-ul este creat, unde conținutul și funcționalitățile sunt organizate. După adăugarea de culori, iconițe, imagini, logo-uri obținem mockup-ul iar după adăugare elementelor interactive, obținem prototipul.

Schiță – reprezentare a aplicației cu precizie scăzută, poate fi făcută pe o foaie de hârtie sau într-un program specializat, folosite pentru a da formă ideilor venite în urma unei sesiuni de brainstorming

Wireframe – reprezentare a aplicației cu precizie scăzută, echivalent al unui schelet al aplicației, folosit pentru a organiza și ordona conținut și funcționalități, nu include culori, font-uri etc.

Mockup - reprezentare a aplicației cu precizie medie, implică adăugarea de culori, font-uri, conținut text, imagini, logo-uri

Prototip - reprezentare a aplicației cu precizie ridicată, folosește ca bază mockup-ul la care adaugă concepte de UX, interacțiuni, animații; oferă senzația de folosirea a aplicației finalizate dar nu conține logică/funcționalitate, ci doar imagini conectate între ele

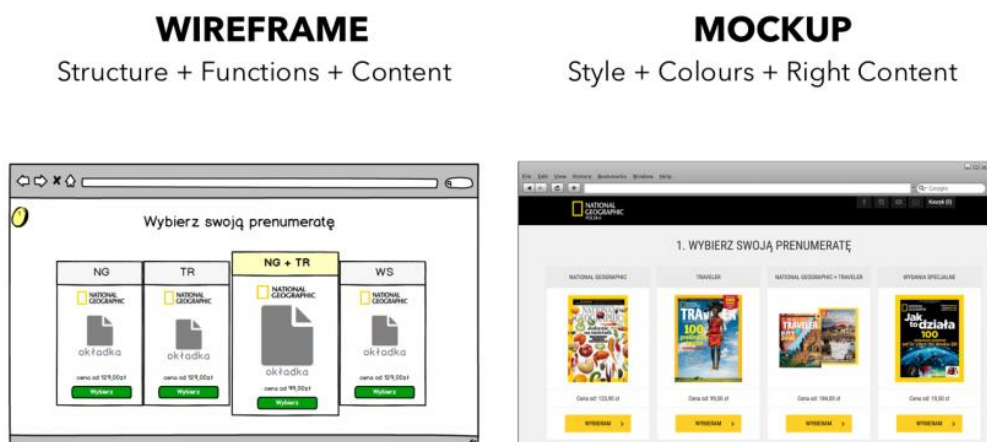


Figura 5 - Rezumat al procesului de UI design

(Warcholinski, 2016)

UI Development

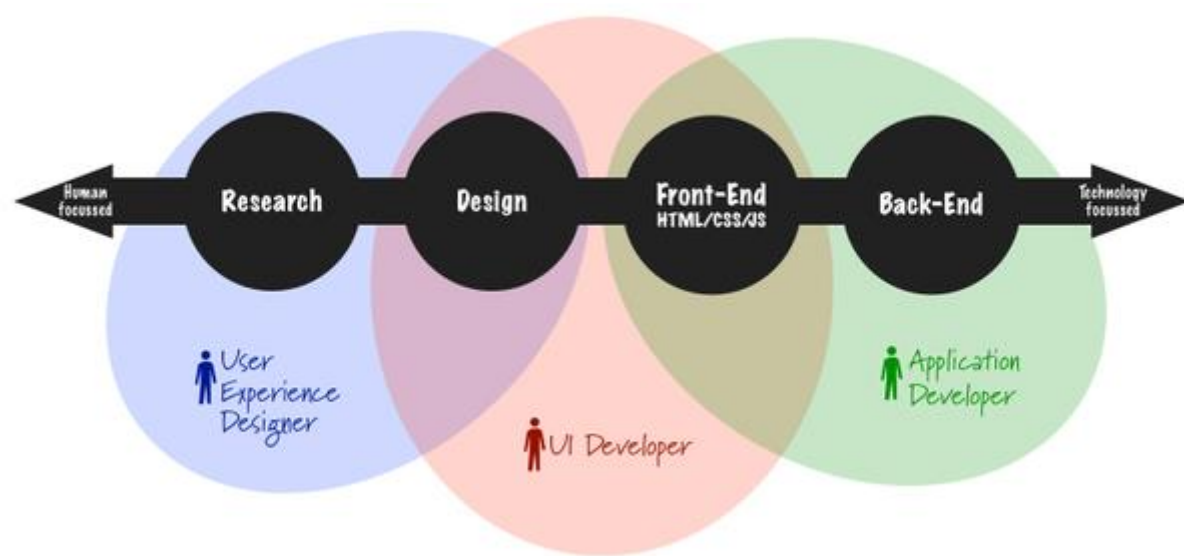


Figura 6 - Vedere de ansamblu asupra procesului de dezvoltare a unei aplicații - de la idei, concepte și design spre implementarea propriu-zisă

Un UI/UX Designer combină cercetarea și abilitățile de design pentru a înțelege nevoile utilizatorului și produce concepte/soluții/design-uri. Accentul cade pe comportamentul uman și psihologie și implică creativitate în rezolvarea problemelor.

Un Application Developer este în general persoana care face produsul să funcționeze, folosind cod, logică și creează o interfață sumară, doar pentru a ilustra acele funcționalități.

Persoana de legătură, care se asigură că implementarea în cod este în concordanță cu design-ul se numește UI Developer (Figura 6 - Vedere de ansamblu asupra procesului de dezvoltare a unei aplicații - de la idei, concepte și design spre implementarea propriu-zisă). Aceasta combină cunoștințe de design cu cele tehnice, realizează un produs care arată bine dar și funcționează și înțelege foarte bine comportamentele de redare ale browser-elor.

(Melbourne, 2011)

SCSS

În dezvoltarea unei aplicații web, tehnologiile folosite pe latura Front-end sunt HTML, CSS și un limbaj ce oferă logică și conținut dinamic. La fel cum HTML a evoluat de-a lungul timpului, renunțând la etichete învechite și adăugând altele noi, unele urmărind să înlocuiască tehnologia Flash (HTML5), CSS a ținut pasul și a introdus variabile și proprietăți noi (CSS3).

Cu toate acestea, s-a simțit în continuare lipsa organizării și repetabilitatea codului, mai ales în aplicațiile mari.

Syntactically Awesome StyleSheets intră în peisaj și oferă dezvoltatorilor UI soluții pentru aceste probleme.

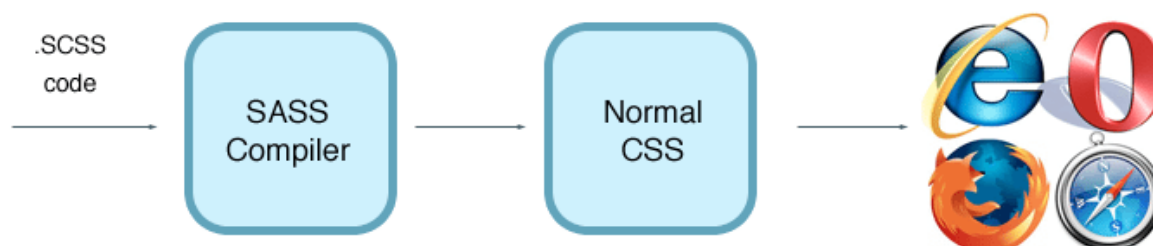


Figura 7 - Procesul de compilare SCSS

(The Power of Sass and Why You Should Embrace CSS Preprocessors, 2017)

SCSS - Avantaje

Preprocesare – din cauza faptului că fișierele CSS pot deveni extrem de mari, complexe și greu de întreținut, SASS oferă o experiență mai plăcută dezvoltatorului în timp ce acesta scrie, întreține sau adaugă cod. În final, SASS va genera CSS obișnuit ce va fi interpretat de browser (Figura 7 - Procesul de compilare SCSS).

Variabile – pot fi declarate global sau local, sporesc consecvența prin reutilizare iar la compilare variabila este înlocuită cu informația stocată în ea

Imbricare/Nesting – simularea ierarhiei HTML într-o formă CSS, oferă o vedere mai clară asupra structurii elementelor, reduce repetabilitatea

Fișiere parțiale – fișiere SCSS prefixate cu linie de subliniere care *nu* vor fi compilate spre CSS ci se comportă ca referințe (util pentru variabile, funcții, mixin-uri), sporesc modularitatea codului, se folosesc împreună cu directiva de @import

Directiva @Import – importarea fișierelor obișnuite sau parțiale SCSS, fișiere CSS simple. La compilare, se face concatenarea acestora iar apoi sunt servite browser-ului

Mixin-uri – blocuri de cod declarate cu denumire și parametri ce pot fi invocate oriunde în cod, cel mai des folosite pentru a reține prefixele unei proprietăți specifice fiecărui browser

Extinderea/Moștenirea – folosită împreună cu declarații *placeholder* (ce nu vor fi transformate în CSS dar pot fi extinse)

Operatori – suport pentru operatori de bază (+, -, *, /, %)

(Sass Basics, fără an)

Pe lângă acestea, SCSS mai oferă și:

- Posibilitatea folosirii sintaxei preferate de dezvoltator:
 - o SASS, ce este similar limbajului Python în sensul că indentarea dictează unde începe și unde se termină un bloc de cod
 - o SCSS, care este similar CSS-ului și folosește acolade în delimitarea codului
- Tipuri de date suportate de variabile: numere, string-uri, culori, valori booleene, valori nule, liste, hărți(maps), referințe la funcții
- Funcții specifici culorilor (transparentize, adjust-hue, invert, complement etc.)
- Funcții și operații pentru string-uri, liste, map-uri
- Interpolare
- Directive de control (@if, @for, @each, @while)
- Pasarea unui bloc de cod drept parametru al unui mixin prin variabila @content
- Și multe altele

(Sass Documentation, fără an)

SCSS - Dezavantaje

Proiectele mari cu multe pagini ce au nevoie de o interfață uniformă și consecventă pot profita la maxim de funcționalitățile SCSS. Însă, dacă se dorește dezvoltarea unei aplicații mici sau a unui site personal, SCSS poate însemna timp pierdut pe instalare și compilare și poate îngreuna procesul de dezvoltare. Pe lângă acest lucru, clasicul CSS ar putea lansa la un moment dat funcționalități comparabile cu cele oferite de SCSS iar trecerea înapoi la sintaxa CSS ar putea fi una grea.

CSS3 – noutăți și obstacole

CSS3 este ultima versiune a limbajului Cascading Style Sheets și aduce numeroase noutăți: colțuri rotunjite, umbre, gradient, tranziții și animații, precum și tipuri de layout: multi coloană, flex și grid. Tranzițiile și animațiile (tranziții cu mai multe faze decât început și sfârșit) rulează fluid în orice browser modern și oferă un comportament natural elementelor din pagină. Din păcate, unele dintre aceste noi funcționalități nu sunt suportate, au suport redus sau necesită prefixe în browser-e precum Internet Explorer sau Safari.

Aranjarea în pagină a elementelor nu a fost niciodată mai ușoară. Nu mai este nevoie să folosim trucuri cu tabele, float-uri, centrare pe verticală folosind line-height, ci ne bucurăm de ușurința utilizării proprietăților din categoriile flex și grid pentru a alinia (și mai ales centra) pe orice axă dorim. De asemenea, grid-urile create folosind float-uri și margini negative (e.g. Bootstrap), fiind deja predispuse la bug-uri, vor pierde din popularitate odată cu implementarea celei mai noi versiuni de display de tip grid și pe Internet Explorer.

Deși animațiile CSS au impresionat prin versatilitatea lor, acestea pot scădea drastic performanța unei aplicații. Impactul cel mai mare asupra performanței îl au animațiile aplicate pe proprietăți ce țin de poziționare și *box model* (e.g. top, right, width, height, margin, padding, etc.). Animațiile ce sunt “desenate” mai fluid sunt cele aplicate pe proprietatea *transform* (ce permite scalare, rotire, translație în contexte 2D sau 3D) deoarece folosesc accelerarea unității de procesare grafică. În ciuda acestui lucru, abuzarea de animații sau setarea lor pe rulare infinită va afecta performanța oricărei aplicații din cauza limitărilor hardware a device-ului folosit de client.

(Matyus, fără an)

Front-end

Tehnologiile abordate pe latura front-end sunt Angular și Typescript.

Angular este un framework Javascript ce creează SPAs (single page applications) și a câștigat popularitate datorită faptului că device-ul utilizatorului a devenit din ce în ce mai performant, încât nu a mai fost nevoie ca serverul să preia toată procesarea. Astfel paginile sunt încărcate dinamic, fiind generat cod nou HTML, CSS și Javascript pentru fiecare componentă încărcată.

Datorită faptului că versiunile de Angular au suferit numeroase modificări (unele majore precum trecerea de la versiunea 1 la 2) iar proiectele *boilerplate* (proiecte deja create, cu unele dependențe instalate ce ar trebui să ruleze *out of the box*) găsite pe internet nu funcționau mereu cum trebuie și din dorința de standardizare, a apărut Angular-CLI. Astfel, cu ajutorul unor comenzi simple, se pot crea și actualiza proiecte.

Odată cu crearea de proiecte, Angular-CLI instalează plugin-uri, printre care și câteva extrem de folositoare în dezvoltarea stilului:

- Preprocesorul SCSS – acesta se instalează odată cu folosirea parametrului “--style=scss” la crearea unui nou proiect
- Postprocesorul Autoprefixer – plugin ce permite dezvoltatorului să scrie proprietăți CSS fără a adăuga prefixe de care au nevoie anumite browser-e; acest plugin parcurge CSS-ul generat și adaugă prefixe în concordanță cu <https://caniuse.com>
- Postprocesorul Cssnano – plugin pentru compresarea CSS-ului prin eliminarea spațiilor și comentariilor

Limbajul Typescript folosește Javascript la bază și vine cu câteva îmbunătățiri:

- Posibilitatea de folosire a tipurilor explicite de date pentru variabile sau metode pentru evitarea erorilor ce pot apărea la rulare (sintaxa obișnuită de Javascript este validă și în Typescript)
- IDE-urile (Integrated Development Environment) pot:
 - oferi feedback în timp real asupra codului defectuos
 - oferi sugestii de metode sau attribute la tastare
 - sugera funcții ce aparțin unor librării


Aspecte practice

User experience

Aplicația, în stadiul curent, nu atinge toate atributele specifice unui bun UX deoarece nu dorește să fie decât un “proof of concept”. Însă, ne putem imagina că există o versiune finalizată a aplicației care utilizează date reale și comunică cu un server și o bază de date pentru a-și îndeplini scopul. Prin urmare putem spune că aplicația ar putea fi:

- valoroasă (deoarece nu mai există un produs similar destinat studenților facultății și deoarece satisface majoritatea nevoilor studenților)
- ușor de găsit (presupunând că va fi la fel de ușor de găsit de către motoarele de căutare precum actualele site-uri ale facultății iar funcționalitățile aplicației sunt ușor de reperat)
- folositoare (studenții vor putea urmări știrile facultății, notele, absențele și taxele, notificările de emailuri etc.)
- utilizabilă (din nou, presupunând că versiunea completă există și își face treaba bine, fără ca server-ul să fie indisponibil sau alte probleme tehnice)
- dezirabilă (prin interfața sa prietenoasă și vie)
- accesibilă (pe desktop dar și pe ecrane mobile)
- credibilă (prezentând sigla oficială în concordanță cu ghidul brandului UAIC).

Pentru a avea o idee clară asupra audienței țintă, am cumulat experiența proprie cu o persona reprezentată în Tabela 1 - Persona folosită pentru aplicația FIIDashboard.

PERSONA	Student în anul 3
POZĂ	
NUME FICTIV	Ionescu Claudia

FUNCȚIE/RESPONSABILITĂȚI	<ul style="list-style-type: none"> - A fi la curent cu noutățile din cadrul universității/facultății pentru a ridica adeverințe, semna contracte, plăti taxe etc. - A plăti taxele la timp - A învăța pentru seminare și examene - A obține note de trecere
DETALII DEMOGRAFICE	<ul style="list-style-type: none"> - Sex: feminin - Vârstă: 21 de ani - Stare civilă: necăsătorită - Naționalitate: română
SCOP ȘI ACTIVITĂȚI	<ul style="list-style-type: none"> - Verifică în mod regulat paginile profesorilor și webmail-ul facultății pentru noutăți - Atunci când caută informații dorește să le găsească cât mai rapid - Folosește intens aplicații de socializare pe telefonul mobil
DORINȚE ȘI FRUSTRĂRI	<ul style="list-style-type: none"> - Să poată găsi și citi cursuri pe telefonul mobil fără să fie oprită de parole pe care le uită mereu - Să poată citi mai ușor orarul pe telefonul mobil - Să primească notificări în timp real pentru anunțuri importante

Tabela 1 - Persona folosită pentru aplicația FIIDashboard

Din punct de vedere al conținutului de tip text, am încercat să folosesc cuvinte sugestive pentru link-uri și butoane iar mesajele de validare și pagina de “404 - pagina nu a fost găsită” oferă mesaje ajutătoare ce nu critică utilizatorul pentru greșelile sale ci îl îndrumă în direcția corectă (spre exemplu, la completarea greșită a unui câmp de email, se vede un mesaj de eroare ce ulterior oferă o sugestie de completare – “textul introdus respectă tiparul nume@domeniu.com?”).

În continuare voi aduce la lumină câteva probleme ale paginii actuale eSims, din punct de vedere UX. Logarea în contul studentului/profesorului este principala funcționalitate a site-ului, însă butonul de logare este greu de reperat, există simboluri care creează confuzie (ghilimelele «») iar logo-ul universității nu a fost reactualizat în conformitate cu ghidul brandului. De ce ar fi importantă ultima problemă menționată? Deoarece creează confuzie (spre exemplu, la găsirea pe internet a unei pagini similare Facebook, ce folosește logo-ul său dar cu mici modificări de font și culoare, audiența ar deveni nedumerită și ar putea avea suspiciuni cum că se află pe un site de phishing – înșelăciune electronică care constă în

obținerea unor date confidențiale precum credențiale sau date bancare prin manipularea identității unei persoane sau instituții).

LOGIN

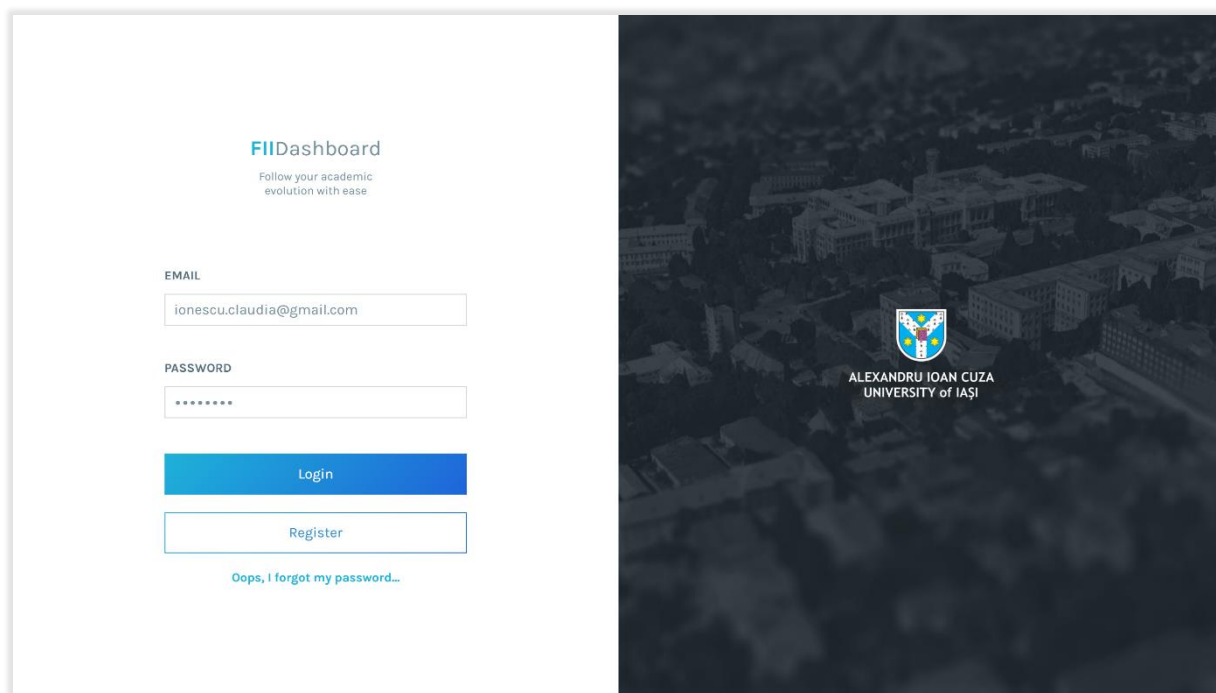


Figura 8 – Logarea FIIDashboard pe desktop

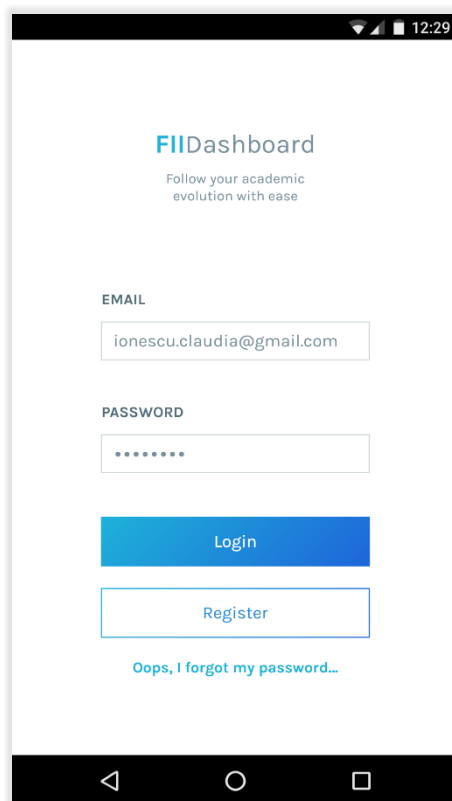


Figura 9 – Logarea FIIDashboard pe mobil

Am creat o pagină de start ce va cere în primul rând logarea utilizatorului (Figura 8 – Logarea FIIDashboard pe desktop, Figura 9 – Logarea FIIDashboard pe mobil), prezentând

titlul și scopul aplicației pe scurt. Drept câmpuri ce trebuie completate am optat pentru email și parolă din cauza faptului că nici un student nu își cunoaște numărul matricol pe de rost. Pentru a facilita completarea datelor, am introdus mesaje de validare ajutătoare. Butonul de logare captează prima oară atenția utilizatorului datorită fundalului său viu colorat iar dacă se dorește înregistrarea unui nou cont, atenția va cădea asupra butonului de înregistrare aflat dedesubtul celui de logare. În josul paginii se poate găsi și un link spre resetarea parolei în cazul uitării acesteia. Prin această primă pagină am comunicat utilizatorului câteva elemente ușor de reperat în restul aplicației: înfățișarea input-urilor, mesajelor de eroare/validare și gradul de importanță al butoanelor.

LAYOUT

Odată logat în aplicație, utilizatorul este întâmpinat de o pagină ce conține 3 zone majore: meniul din stânga, antetul și zona dinamică a aplicației.

LEFT SIDE

În zona de meniu (Figura 10- Pagina de profil (desktop)) se poate vedea titlul aplicației/logo-ul. Deoarece titlul este inițial informația cea mai importantă a paginii iar majoritatea utilizatorilor citesc de la stânga la dreapta, logo-ul a fost plasat în colțul din stânga sus. Imediat dedesubt a fost plasată o confirmare a identității utilizatorului logat (feedback primit după logare), reprezentată prin poza de profil și câteva detalii (nume, an și grupă). Imaginii de profil îi este asociată un indicator/signifier: o iconiță de editare ce va duce la pagina de profil.

PROFILE

Se oferă utilizatorului posibilitatea de a-și schimba poza de profil, e-mail-ul și numărul de telefon (deoarece acestea se schimbă cel mai des), restul informațiilor fiind doar afișate/read-only. Consecvența se observă prin repetarea elementelor precum input-uri, mesaje de validare, butoane în aceeași manieră ca pe pagina de logare.

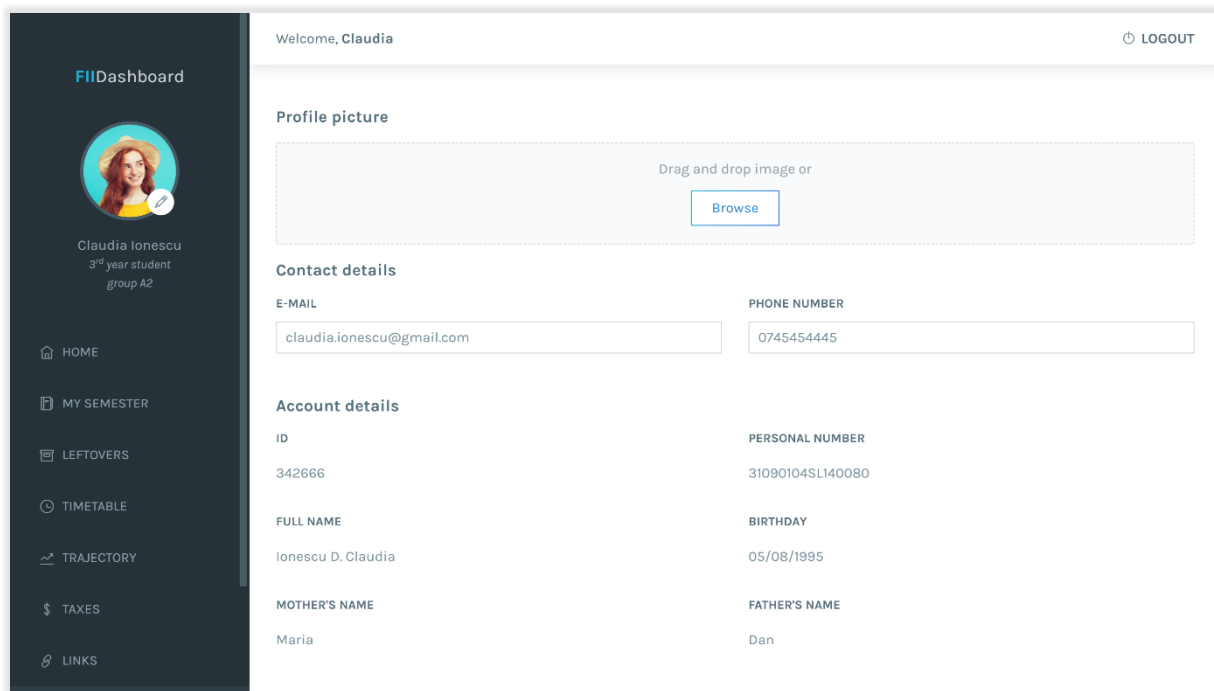


Figura 10- Pagina de profil (desktop)

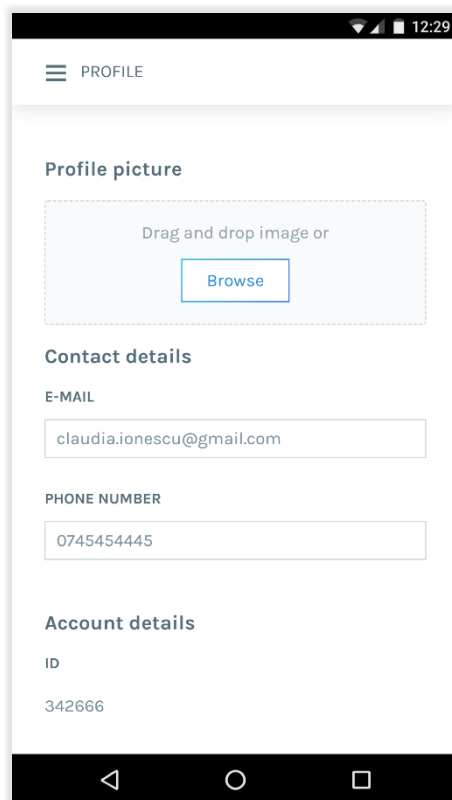


Figura 11- Pagina de profil (mobil)

MENU

Tot în partea stângă am plasat meniul (Figura 12 - Meniul implementat) sub formă de acordeon și cu un număr acceptabil de link-uri, compactând pe cele similare într-un sub-meniu. Se oferă feedback utilizatorului prin afișarea unei benzi albastre lângă linkul spre

pagina curentă și se oferă un semnificativ/indicator spre sub-meniu (dacă acesta există) printr-o săgeată ce indică “jos”. Dacă se dorește închiderea sub-meniului, săgeata va fi direcționată în sus pentru a indica colapsarea acestuia. Acești indicatori se pot vedea la plasarea cursorului deasupra unui item din meniu. Animațiile, atât rotirea săgeții cât și expandarea/colapsarea sub-meniurilor, oferă feedback printr-o mișcare naturală, spre deosebire de schimbarea/apariția lor bruscă.

Sub-meniurile au fost folosite pentru obiectele din semestrul curent sau semestrele trecute și folosesc color-coding: pentru a repera mult mai ușor obiectele ce prezintă interes, au fost asignate 6 culori la cele 6 obiecte per semestru. Astfel utilizatorul poate naviga mai rapid în aplicație pe baza culorilor odată cu memorarea lor, nemaifiind nevoie să citească integral titlurile. Dacă studentul repetă obiecte din ani trecuți, cele 6 culori se vor repeta deoarece introducerea mai multor culori în aplicație poate fi copleșitor pentru utilizator.

Deoarece majoritatea studenților posedă un telefon mobil de tip smartphone, am decis ca aplicația să se adapteze nevoilor acestor studenți. Astfel, pentru browser-e rulând pe device-uri mici, meniul este ascuns și poate fi activat prin butonul *hamburger* (simbol universal pentru meniu) din colțul stânga sus. Acesta va ocupa tot ecranul pentru lizibilitate maximă și poate fi închis prin butonul de închidere din dreapta sus. Pe varianta pentru mobil se observă și opțiunea de delogare la finalul meniului.

HEADER

În partea de sus a ecranului am afișat un mesaj de „bun-venit” iar în partea dreaptă se află link-ul pentru delogare, ce a fost plasat în colț deoarece seamănă cu acțiunea de închidere a unui element (fereastră, modal etc.). Header-ul este fixat în partea de sus a ecranului pentru a avea acces imediat spre delogare sau spre meniul afișat pe mobil.

Dedesubtul antetului se află zona dinamică a site-ului, aceasta fiind populată cu alte elemente odată cu schimbarea paginii.

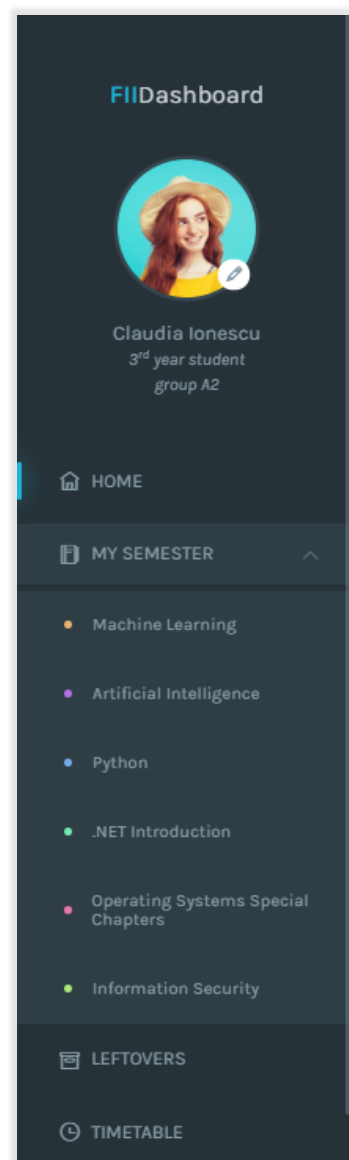


Figura 12 - Meniul implementat

HOME PAGE

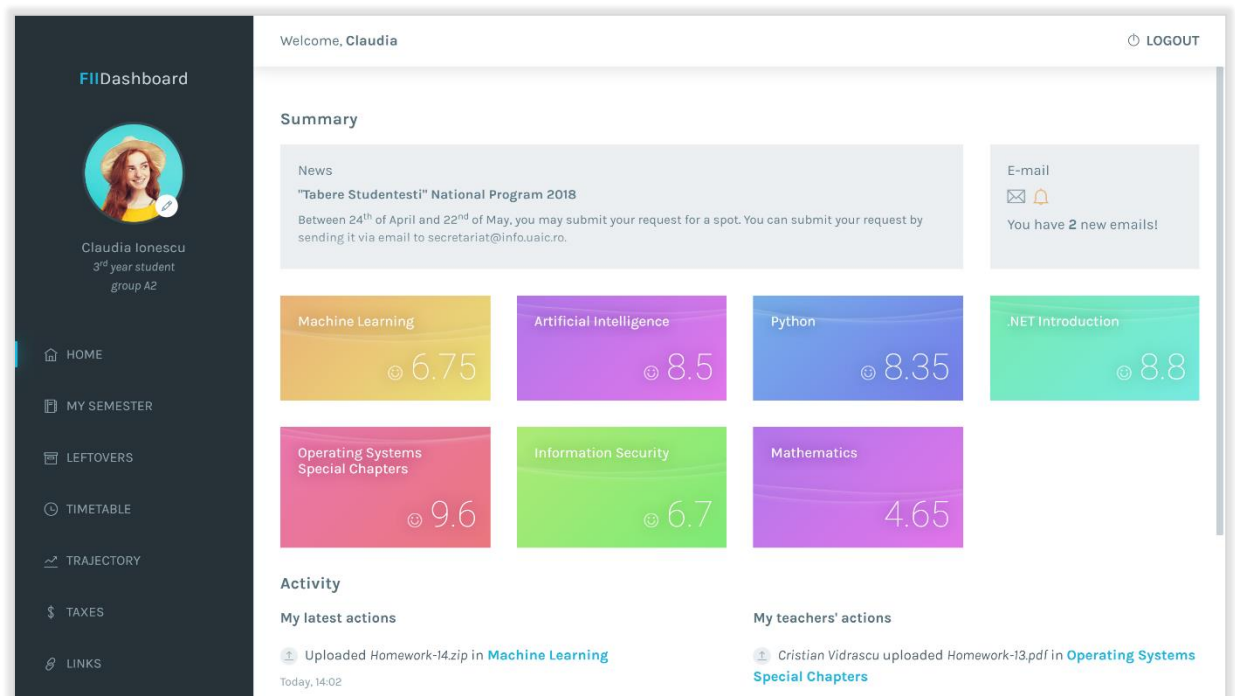


Figura 13 - Pagina de start implementată (desktop)

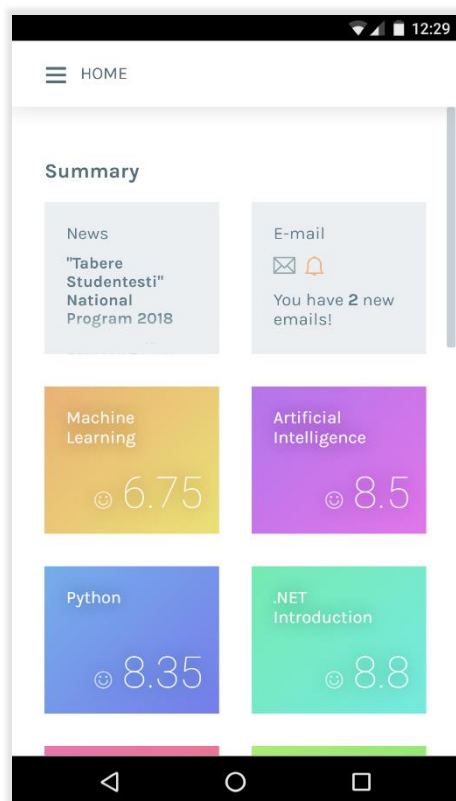


Figura 14- Pagina de start implementată (mobil)

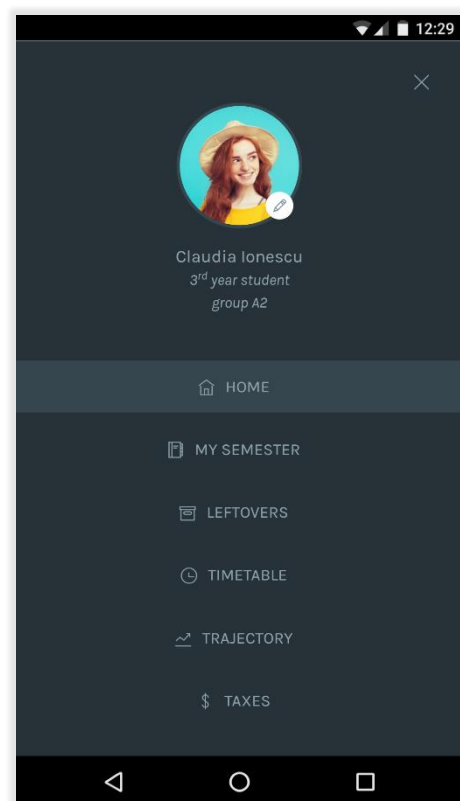


Figura 15 – Meniul pe mobil

Pagina de start (Figura 13 - Pagina de start implementată) oferă scurtături către cele mai accesate zone ale aplicației. Așadar, putem observa următoarele arii de interes:

- News – zonă ce va arăta ultimul anunț important scris de angajații facultății.
- Email – zonă ce indică utilizatorului câte email-uri necitite are. Iconița imită prin animația sa mișcarea unui clopot real și va apărea doar dacă există email-uri noi
- Tile-uri pentru obiectele studiate în semestrul curent cât și cele ce trebuie refăcute. Am refolosit color-coding pentru a repera imediat obiectul de interes pe baza culorii acestuia. Pentru a oferi o vedere de ansamblu, am introdus animația unor valuri, unde nivelul acestora indică subtil promovabilitatea. Nota momentan obținută este însoțită de o față zâmbitoare atunci când valoarea sa întrece valoarea 5. Notele sunt însoțite la acțiunea de *hover* de un mesaj ce confirmă situația curentă (e.g. „You passed!”). Fiecare dintre aceste tile-uri direcționează la click spre pagina disciplinei.
- Activity – secțiune a paginii ce oferă un istoric al acțiunilor studentului în aplicație dar și ultimele acțiuni efectuate de cadrele didactice (încărcarea de fișiere, teme etc.). Aceste log-uri sunt însoțite de link-uri spre pagina în care s-au făcut modificări, cât și de date temporale.

TIMETABLE

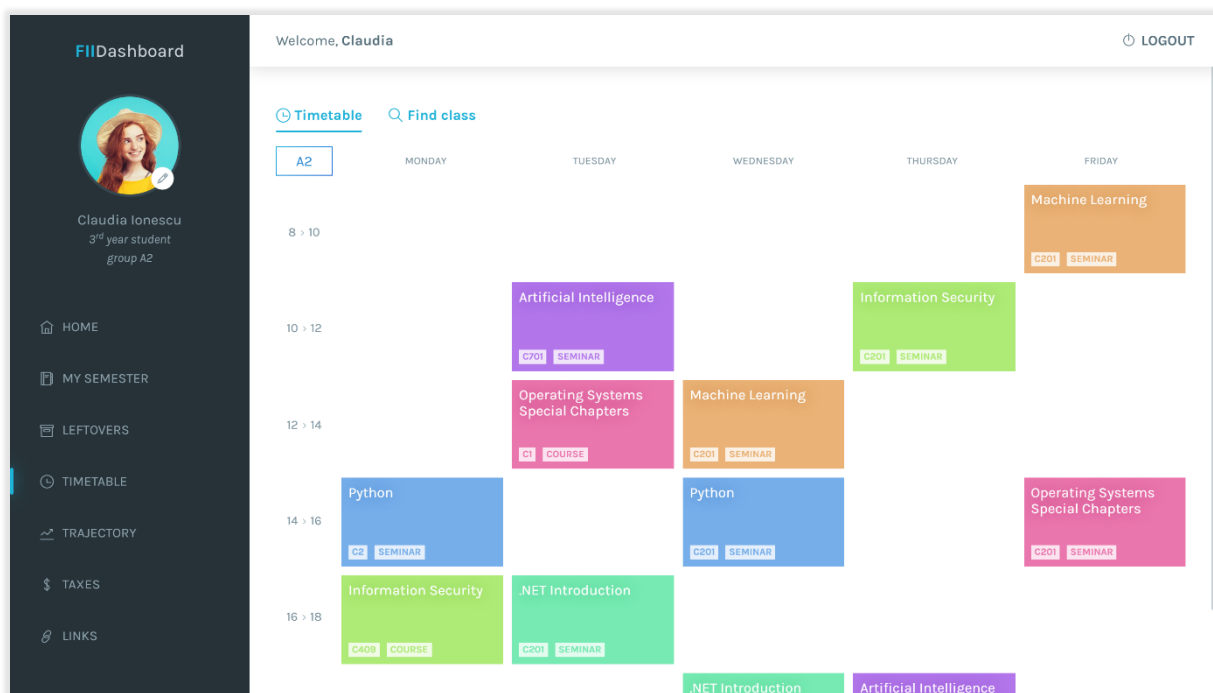


Figura 16 - Orarul pe desktop

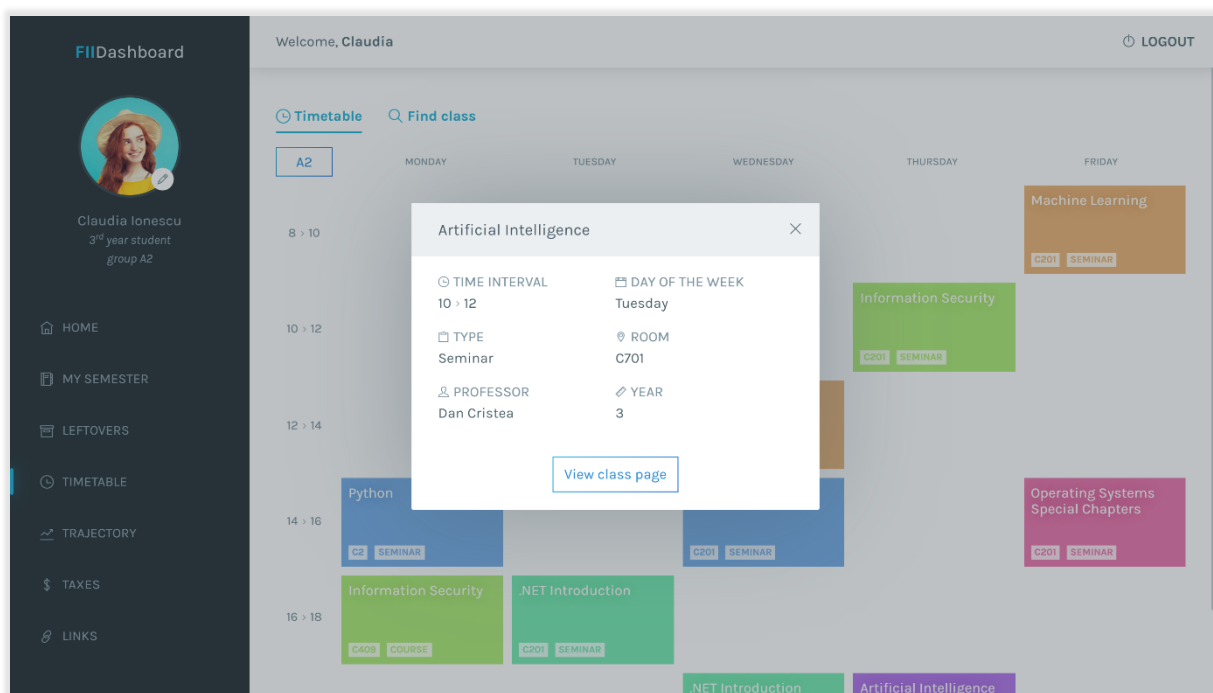


Figura 17 - Orarul pe desktop, modal conținând detalii

Welcome, Claudia

LOGOUT

Timetable Find class

Subject	Professor	Type	Location	Time	Group	Day	Year
<input type="text" value="Subject name"/>	<input type="text" value="Professor name"/>	<input type="text" value="Course name"/>	<input type="text" value="Room name"/>	<input type="text" value="HH:MM -"/>	<input type="text" value="Group name"/>	<input type="text" value="Day of the week"/>	<input type="text" value="Academic year"/>
Python	Dragos Gavrilit	Seminar	C2	10 - 12	A1	Monday	3
Artificial Intelligence	Dan Cristea	Course	C409	12 - 14	A1	Monday	3
Information Security	Leahu Ioana	Seminar	C701	16 - 18	A1	Monday	3
.NET Introduction	Florin Olariu	Course	C1	8 - 10	A1	Tuesday	3
Python	Dragos Gavrilit	Course	C1	14 - 16	A1	Tuesday	3
Operating Systems Special Chapters	Cristian Vidrascu	Seminar	C201	16 - 18	A1	Tuesday	3
Machine Learning	Liviu Ciortuz	Seminar	C201	18 - 20	A1	Tuesday	3
Operating Systems Special Chapters	Cristian Vidrascu	Seminar	C201	8 - 10	A1	Wednesday	3

Figura 18 - Filtrarea orarului global



Figura 19 - Orarul pe mobil

Una din paginile cele mai frecventate de studenți este cea a orarului (Figura 16 - Orarul pe desktop, Figura 17 - Orarul pe desktop, modal conținând detalii, Figura 19 - Orarul pe mobil).

Am folosit în continuare cele 6 culori aferente disciplinelor și am reorganizat informațiile într-un mod plăcut vizual. La intersecția zilei săptămânii cu intervalul orar se află un element cu titlul disciplinei și cele mai relevante detalii: locația și tipul (curs/seminar). La acțiunea de click ne întâmpină un modal cu mai multe detalii precum numele profesorului și anul de studiu iar în josul modalului se află un buton scurtătură spre pagina disciplinei.

Butonul ce indică grupa selectată are și rolul de a schimba grupa ce poate fi vizualizată. Dacă se dorește o căutare avansată asupra tuturor orarelor, se poate folosi tab-ul "Find class" (Figura 18 - Filtrarea orarului global) ce va încărca un tabel detaliat cu următoarele filtre posibile: numele disciplinei, numele profesorului, tipul disciplinei, locația, intervalul orar, grupa, ziua din săptămână și anul de studiu, toate oferind la tastare sugestii relevante de completare.

Aceleași funcționalități sunt păstrate și pe versiunea de mobil dar cu câteva modificări cerute de lipsa de spațiu: numele disciplinei s-a transformat în acronimul acesteia iar detaliile aferente se observă numai în modal.

SUBJECT

Pe pagina disciplinei (Figura 21 - Pagina disciplinei (mobile), Figura 22 - Pagina disciplinei (mobile) - secțiunea teme) putem ajunge fie din meniul din stânga (Current semester sau Leftovers pentru obiectele ce sunt repetate), fie din modalele găsite în orar, fie apăsând pe unul din tile-urile de pe pagina de start.

Toate paginile disciplinelor urmăresc același șablon:

- Titlul disciplinei
- Profesorii aferenți
- Media finală, nota din examen și media aritmetică a seminarului
- Ultima notă adăugată, însoțită de un buton către istoricul notelor
- Secțiunile: cursuri, teme și prezențe

Din dorința de standardizare, am hotărât o regulă de calcul a mediei simple, ușor de implementat și ușor de memorat, fiind aceeași la toate disciplinele: media aritmetică dintre nota obținută la examenul de curs și media notelor de seminar. Tabelul de note din dreapta sus prezintă iconițe de informare în cazul în care studentul este curios cum a fost obținută nota

respectivă.

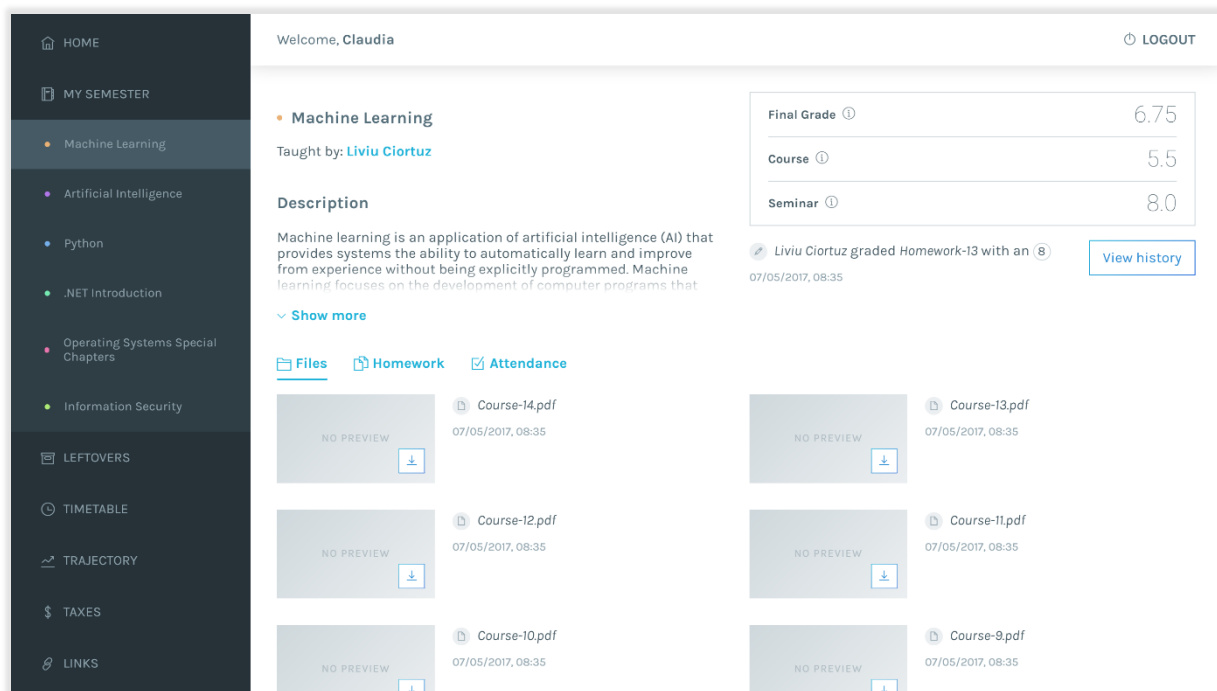


Figura 20 - Pagina disciplinei pe desktop

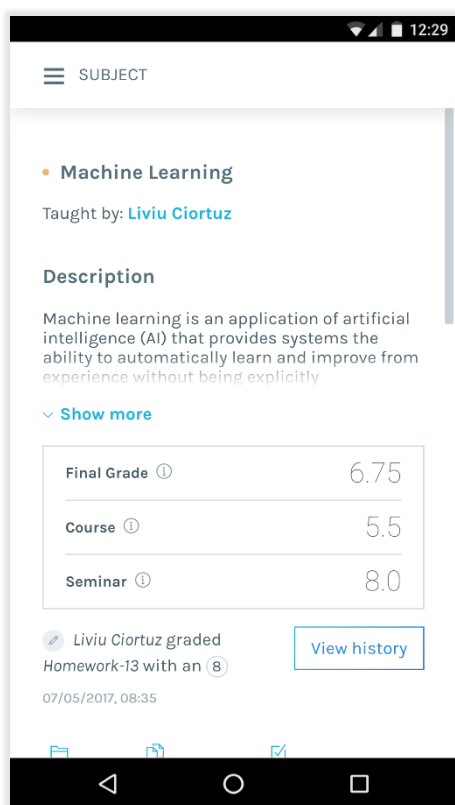


Figura 21 - Pagina disciplinei (mobile)

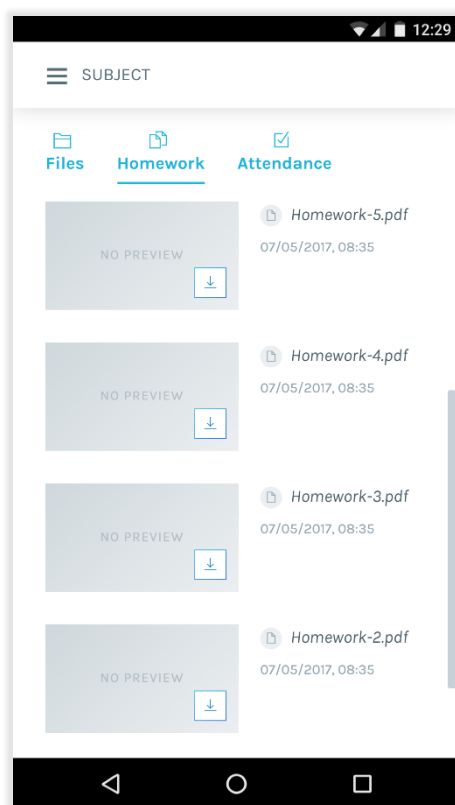


Figura 22 - Pagina disciplinei (mobile) - secțiunea teme

Deoarece studenții nu au acces mereu la evidența prezențelor, am menționat în secțiunea “Attendance” numărul minim de prezențe pentru promovare și numărul de absențe.

TAXES

Pentru a plăti taxe, studenții trebuie să se intereseze care sunt codurile pentru: refacerii activității unei discipline sau plății taxei semestriale, etc. și apoi să le precizeze în timpul plății la bancă. O experiență mai plăcută și mai comodă ar fi simplificarea acestui proces prin ascunderea acestor coduri în spatele unui program ce oferă posibilitatea plății online (vezi Figura 23 - Pagina de plată a taxelor).

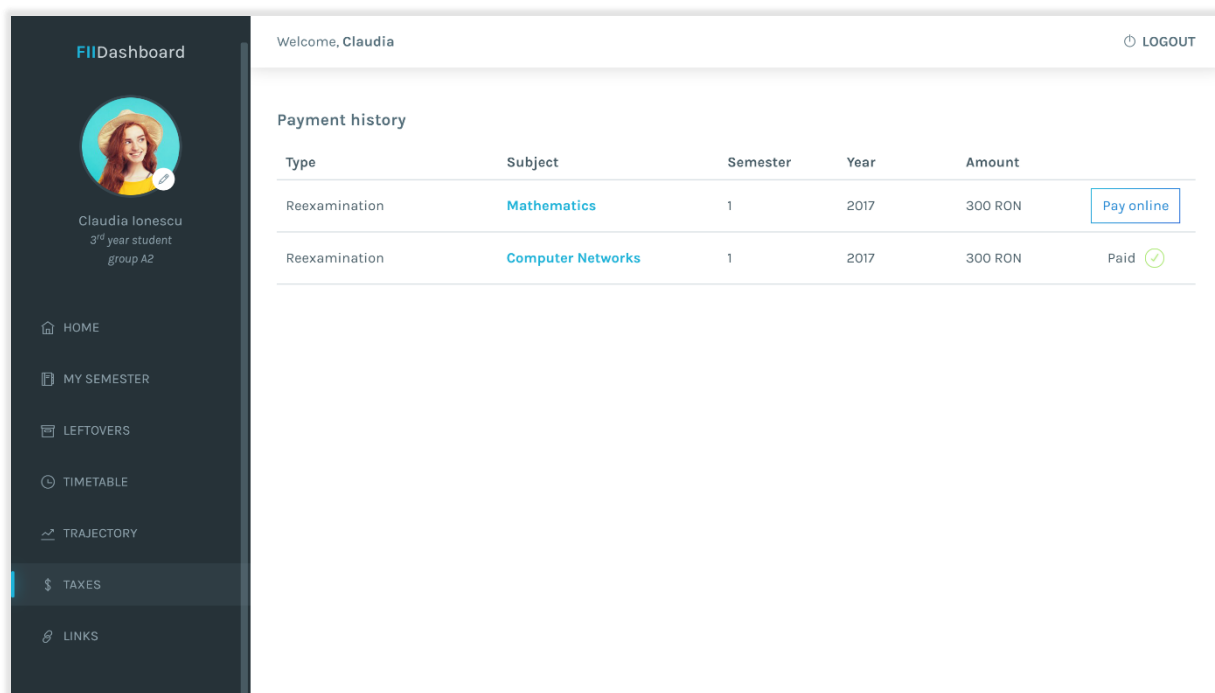


Figure 23 shows the 'Taxes' page in the FII Dashboard. The dashboard header includes 'FII Dashboard', a user profile for 'Claudia Ionescu' (3rd year student, group A2), and a 'Welcome, Claudia' message with a 'LOGOUT' button. The sidebar contains navigation links: HOME, MY SEMESTER, LEFTOVERS, TIMETABLE, TRAJECTORY, TAXES (selected), and LINKS. The main content area displays a 'Payment history' table with the following data:

Type	Subject	Semester	Year	Amount	Status/Action
Reexamination	Mathematics	1	2017	300 RON	Pay online
Reexamination	Computer Networks	1	2017	300 RON	Paid ✓

Figura 23 - Pagina de plată a taxelor

TRAJECTORY

Pentru a avea o vedere de ansamblu asupra traiectoriei educaționale, am introdus o pagină similară celei din eSIMS ce prezintă cei 3 ani de studiu cu detalii precum: nota finală per disciplină, media finală și numărul de credite per semestru (vezi Figura 24 - Pagina traiectoriei studentului cu mediile și creditele obținute de-a lungul celor 3 ani de studiu). În evidență iese anul curent.

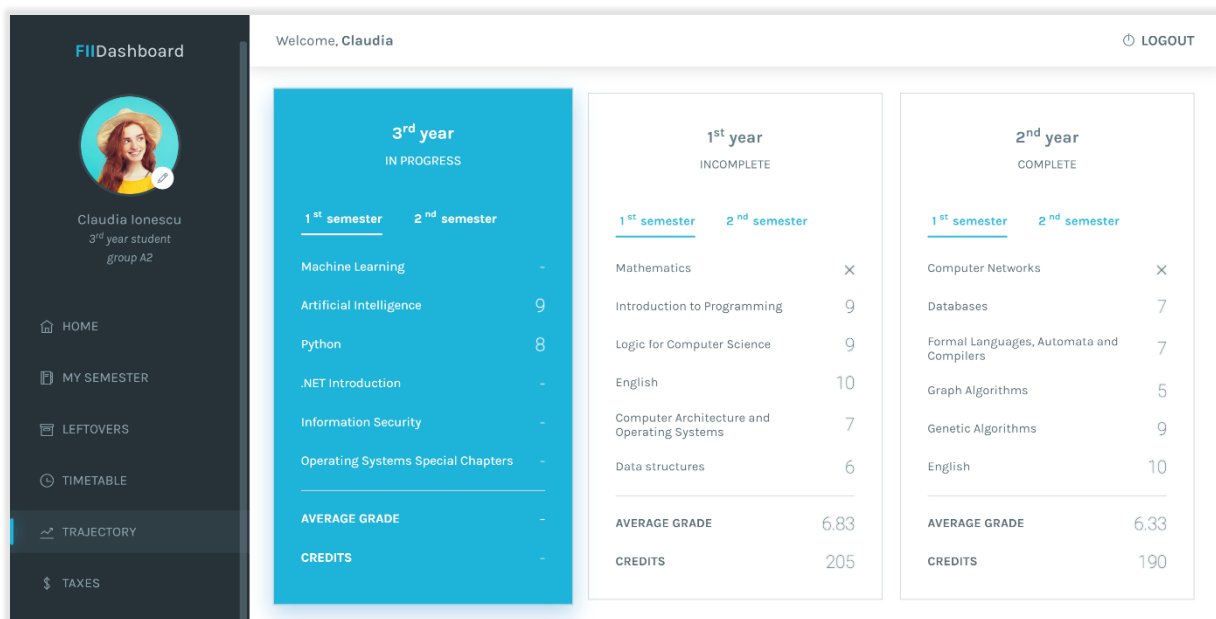


Figura 24 - Pagina traiectoriei studentului cu mediile și creditele obținute de-a lungul celor 3 ani de studiu

La finalul meniului se vor găsi câteva link-uri folositoare precum pagina oficială a Universității iar în cazul device-urilor mobile, se observă link-ul spre delogarea din aplicație.

User interface

Aplicația oferă elemente cunoscute utilizatorului (butoane, input-uri, link-uri etc.) ce se comportă în mod consecvent în orice pagină. Interfața are un aspect curat și aerisit. Simplitatea este dată de limbajul concis, și spațierea elementelor. Informațiile sunt structurate în pagină după importanța lor (e.g. pagina de „acasă” în care găsim știri, notificări, situația generală a disciplinelor și ultimele modificări aferente acestora). Culoarele folosite sunt neutre, gri-urile fiind alese pentru a nu distra atenția utilizatorului de la ce este mai important: elementele ce țin de discipline și folosesc una din cele 6 culori de accent sau albastrul folosit în aplicație pentru a semnaliza acțiuni (butoane, link-uri). Titlurile se diferențiază de textul simplu prin grosimea și mărimea fontului, ceea ce sporește capacitatea de scanare prin conținutul aplicației.

Primii pași în procesul de design sunt cercetarea și stabilirea nevoilor clientului. Utilizând propria experiență cu site-urile facultății și întrebările și nemulțumirile studenților revărsate pe grupurile de Facebook, am stabilit câteva funcționalități de bază.

În continuare am făcut o schiță a aplicației iar apoi am desenat wireframe-ul, scheletul aplicației. Software-ul folosit în acest scop se numește Figma și este o aplicație online ce lucrează cu elemente vectoriale pentru a nu pierde din calitatea design-urilor create.

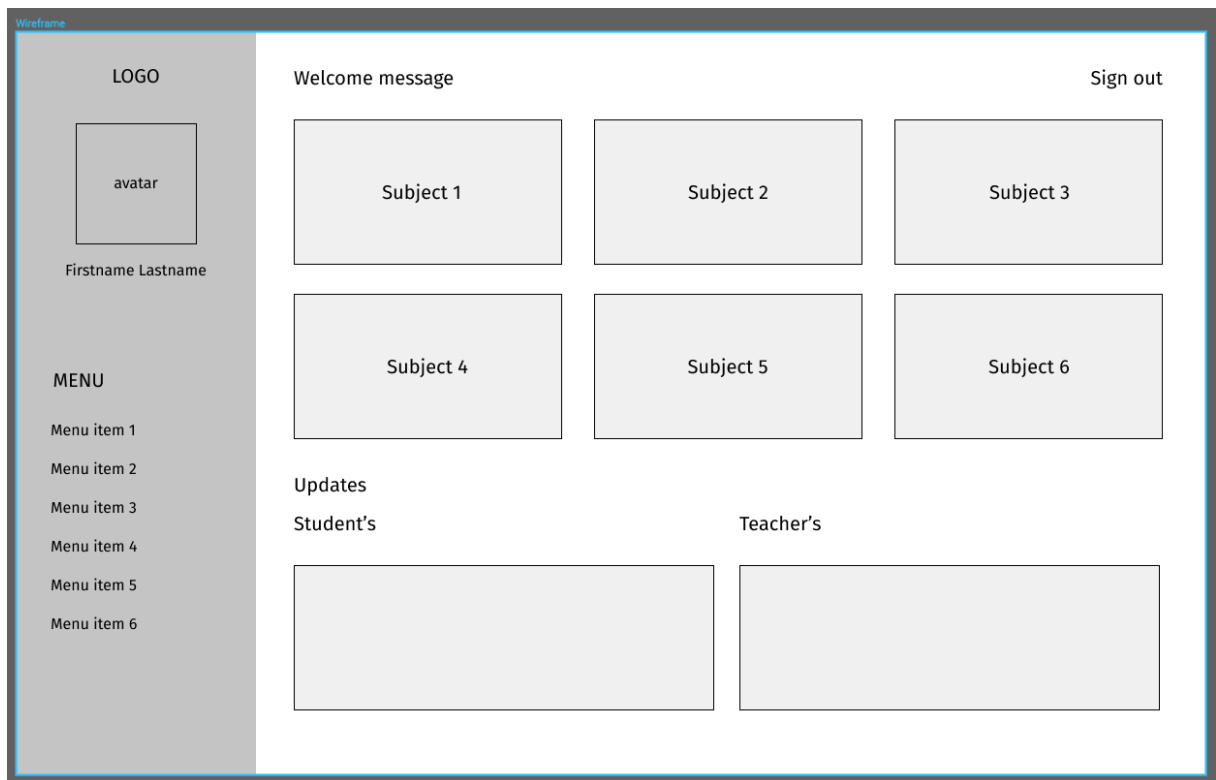


Figura 25 - FIIDashboard - wireframe făcut pentru pagina de start

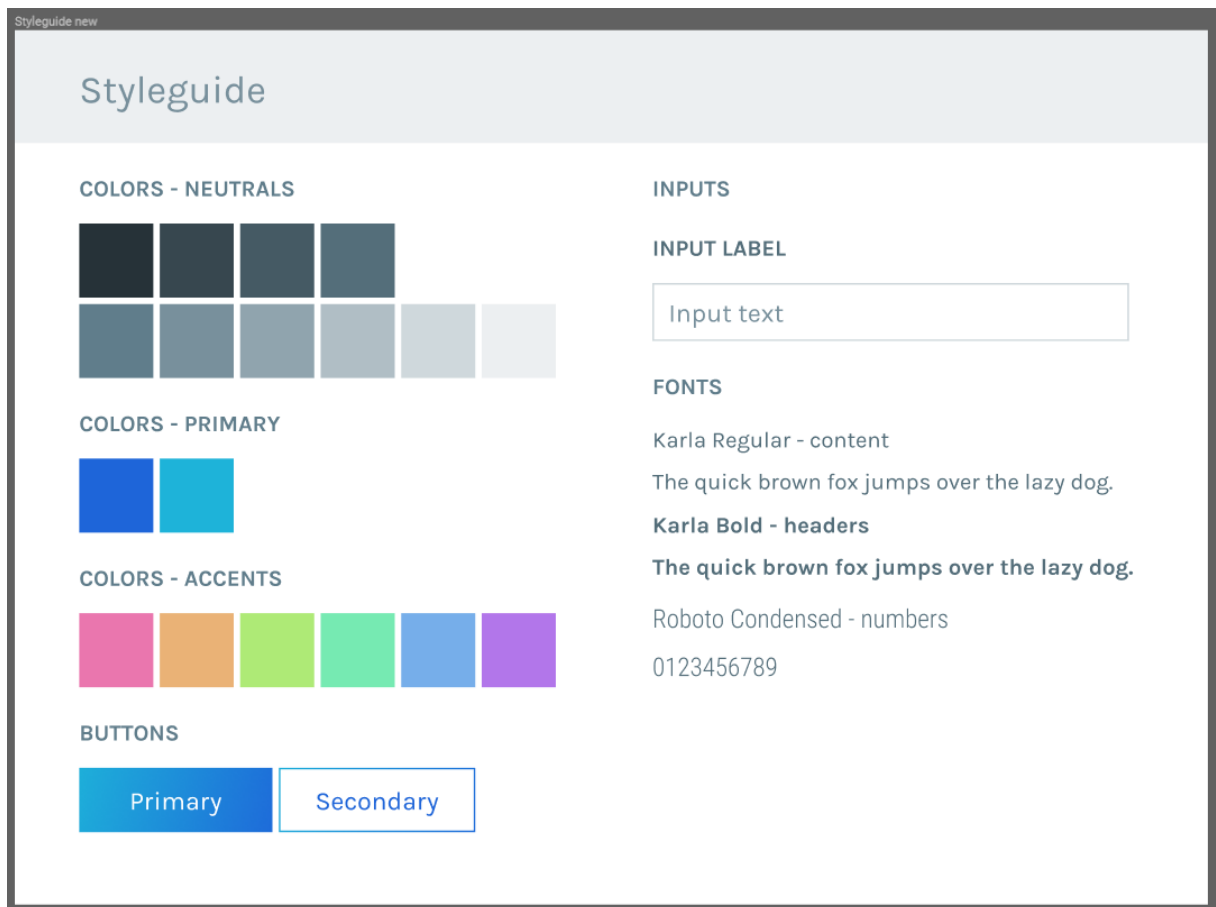


Figura 26 - Scurt Styleguide pentru FIIDashboard – prezentarea unor elemente generice

După efectuarea wireframe-ului (ilustrat în Figura 25 - FIIDashboard - wireframe făcut pentru pagina de start), tot în Figma am creat mockup-ul – adăugarea de culori, fonturi, elemente detaliate. Am creat și un scurt Styleguide (Figura 26 - Scurt Styleguide pentru FIIDashboard – prezentarea unor elemente generice) – un ghid sau un set de reguli ce asigură consecvența și uniformitatea designului.

Primele mockup-uri create au avut în vedere atmosfera generală a aplicației (look and feel). De obicei designer-ul și dezvoltatorul sunt persoane diferite, însă în acest caz am continuat design-ul prin implementarea acestuia și pe parcurs am improvizat și varianta de mobil.

Pentru început am exersat folosirea styleguide-ului în câteva pagini esențiale (pagina de login și cea de start) în Figura 27 - Mockup inițial pentru pagina de login FIIDashboard, Figura 28 - Mockup inițial al paginii de start, Figura 29 - Mockup detaliat pentru meniul aplicației.

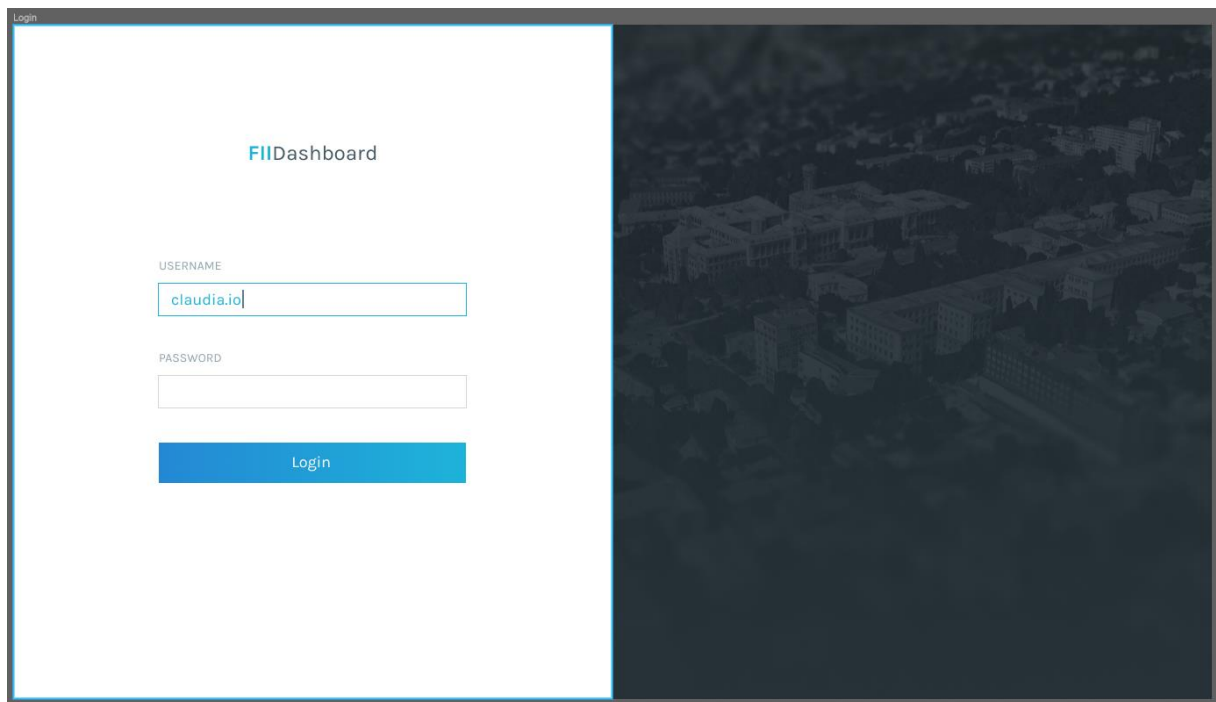


Figura 27 - Mockup inițial pentru pagina de login FIIDashboard

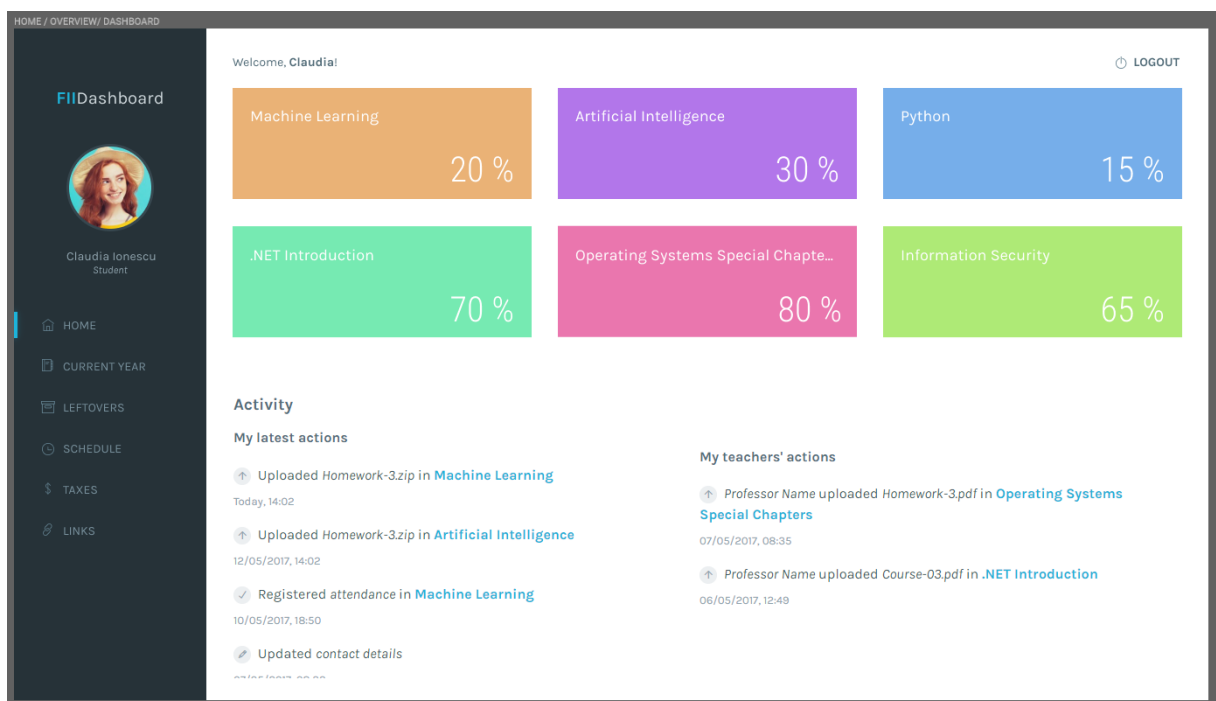


Figura 28 - Mockup inițial al paginii de start

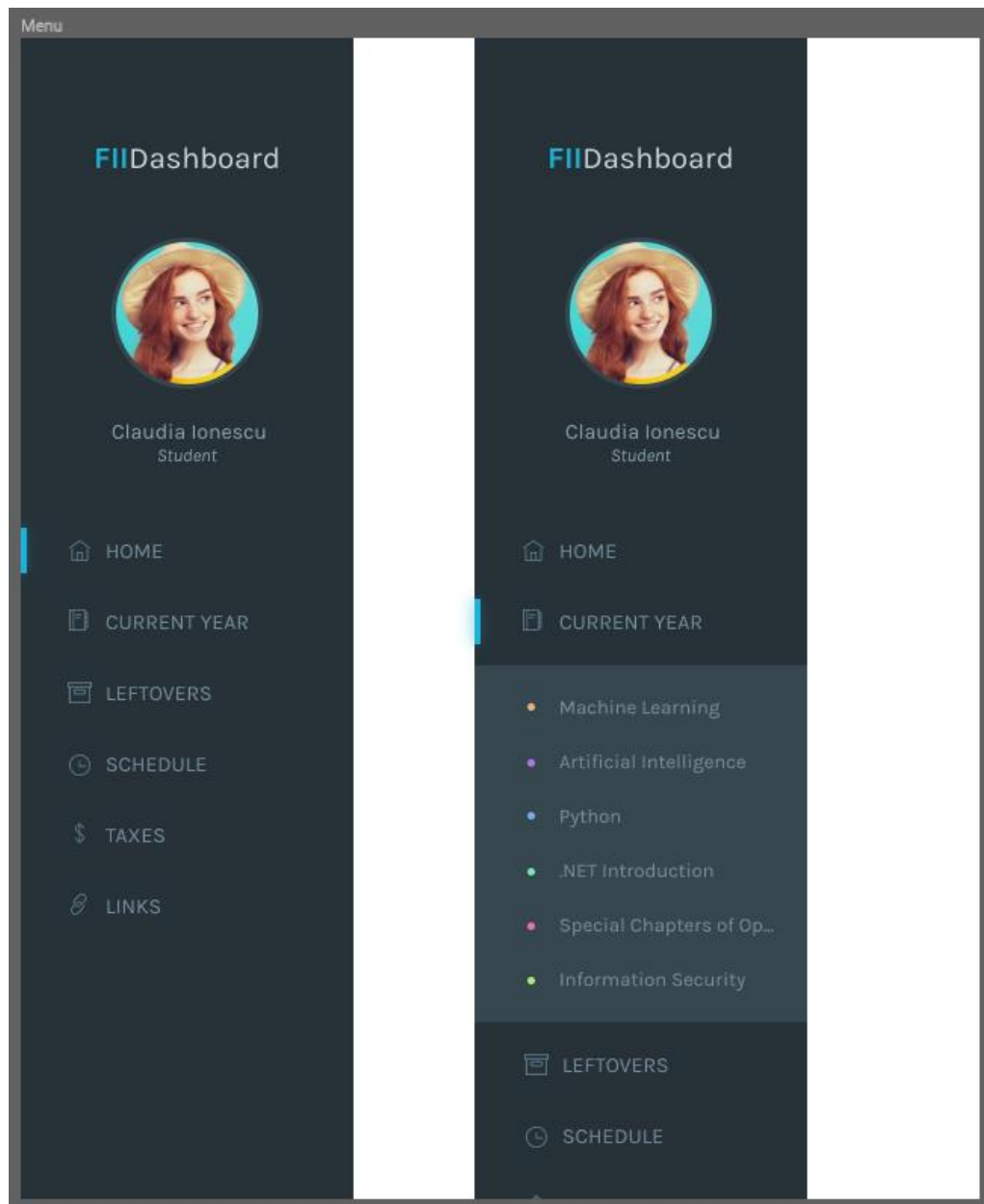


Figura 29 - Mockup detaliat pentru meniul aplicației

În final ar trebui creat prototipul pentru a explica clientului și dezvoltatorilor cum se poate interacționa cu aplicația. Prototipul a fost înlocuit cu o aplicație Angular pentru a oferi mai multă libertate în ceea ce privește interacțiunea.

UI development

Din dorința de a avea un cod organizat și ușor de întreținut, am optat pentru Angular, tehnologie ce permite încapsularea stilului din componentă. Acest lucru înseamnă că stilurile unei componente vor fi aplicate doar acelei componente și nu vor putea modifica elemente din restul paginii. Astfel, dezvoltatorii au o vedere mai bună asupra codului, nefiind nevoiți să lucreze cu fișiere CSS foarte mari.

Deși încapsularea stilului reprezintă un avantaj pentru lucrul cu componente, există și o cantitate mare de stil comun/generic ce trebuie organizat. Nu trebuie să abuzăm de componente dacă structura HTML este extrem de simplă și logica lipsește (e.g. input-urile se repetă în aplicație, însă nu are rost să creăm o componentă, ci vom folosi doar fișiere SCSS).

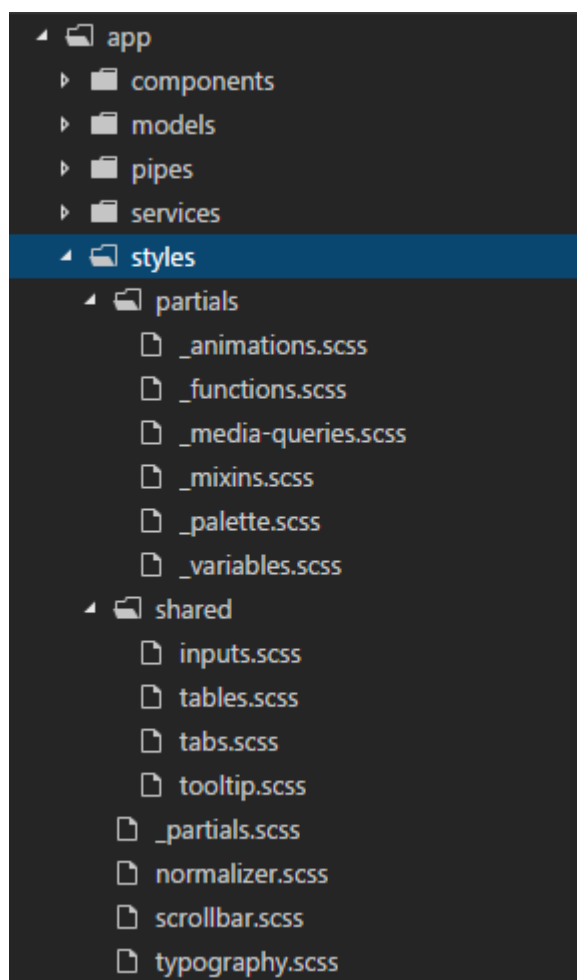


Figura 30 - Arhitectura stilurilor SCSS

După cum se poate vedea în Figura 30 - Arhitectura stilurilor SCSS, pentru a nu aglomera fișierul standard *styles.scss*, am creat un număr de fișiere parțiale (cele prefixate de caracterul *underscore*) iar într-un fișier intitulat „shared” am plasat stil pentru elemente ce ar putea fi componente (dacă ar prezenta logică sau structură complexă).

Printre fișierele parțiale se regăsesc:

- animațiile sub formă de keyframe-uri, responsabile pentru mișcarea de deschidere a sub-meniurilor sau pentru animația de încărcare a paginii etc.
- funcțiile ce generează cod CSS pe baza unor parametri (e.g. funcția *make-boxshadow* va cere ca parametru o culoare, pentru a genera două box-shadow-uri suprapuse cu

nuanțe obținute din parametru), alt exemplu fiind Tabela 2 - Exemplu de funcție - funcție cu nume sugestiv pentru obținerea unei culori dintr-un map

```
@function color($color-name, $color-variant) {
  @return map-get(map-get($palette, $color-name), $color-variant);
}
```

Tabela 2 - Exemplu de funcție - funcție cu nume sugestiv pentru obținerea unei culori dintr-un map

- media-query-urile ce sunt de fapt variabile ce conțin segmente de media-query-uri ce pot fi combinate sau folosite de sine stătător; segmentele folosesc valori prag pentru mobil, tabletă tip portret, tabletă tip peisaj/landscape și desktop (vezi Tabela 3 - Media queries folosite în FIIDashboard)

```
$mobile: 'only screen and (max-width: 599px)';
$tablet-landscape: 'only screen and (min-width: 600px) and (max-width: 899px)';
$tablet-landscape: 'only screen and (min-width: 900px) and (max-width: 1199px)';
$desktop: 'only screen and (min-width: 1200px)';

/* =====
== USAGE EXAMPLE ==

@media #{ $mobile } {
  ...
}

@media #{ $mobile }, #{ $tablet-landscape } {
  ...
}
*/
```

Tabela 3 - Media queries folosite în FIIDashboard

- mixin-urile, care sunt blocuri de cod ce se pot repeta în diverse componente sau ce pot fi aplicate o singură dată global
- paleta de culori ce conține un map împărțit după 3 chei: “primary” - culorile de bază ale aplicației), „accents” - accentele de culoare și „mono” - culorile neutre ale aplicației; accentele au fost generate folosind o culoare de bază iar apoi, prin funcția *adjust-hue* din SCSS, am rotit axa culorile oprindu-mă în 5 puncte ale axei (vezi Tabelul Tabela 4 - Generarea culorilor de accent)

```
$accent-base: #76eab2;
$palette: (
  primary: (
    base: #2488d4,
    light: #1eb3d9,
  ),
);
```



```

    accents: (
      pink: adjust-hue($accent-base, -180%),
      orange: adjust-hue($accent-base, -120%),
      green: adjust-hue($accent-base, -60%),
      cyan: $accent-base,
      blue: adjust-hue($accent-base, 60%),
      purple: adjust-hue($accent-base, 120%),
    ),
    [...]
  );

```

Tabela 4 - Generarea culorilor de accent

Aceste fișiere parțiale au fost adunate într-un fișier numit *partials* ce a fost importat în fișierul *styles*. Astfel, codul va fi transformat în CSS doar la folosirea acestora.

Un fișier important este *normalizer.scss* ce are scopul de a dicta un comportament consistent pe toate browser-ele populare - Chrome, Edge, Firefox, IE 10+, Safari 8+, Opera (e.g. pentru etichetele <sup>, <sub> se va aplica aceeași mărime a fontului și aceeași aliniere pentru toate browser-ele).

(Gallagher, fără an)

Pentru a crea aceeași experiență pe toate browser-ele, am folosit un plugin pentru scrollbar (*ngx-perfect-scrollbar*), stilul fiind dictat de către fișierul *scrollbar.scss* din imagine.

Cât despre directorul *shared*, fișierele dinăuntrul său vor putea fi folosite doar la nevoie (e.g. nu vom importa *inputs.scss* dacă componenta nu folosește input-uri deoarece nu dorim să generăm cod CSS în mod inutil).

Pentru o bună organizare a codului CSS, am folosit o variantă a regulii de ordonare “Outside In” ce oferă o ordine anume și o grupare strânsă a proprietăților înrudite. Pentru a forța această regulă “Outside In” am folosit extensia *CSSComb* ce mi-a permis declararea ordinii dorite și rearanjarea codului la formatare.

În mare, șablonul *CSSComb* pe care l-am folosit este următorul:

- Cuvinte cheie specifice SCSS (variabile, extend, include)
- Proprietăți layout (position, z-index, top, right, display)
- Proprietăți box model (box-sizing, width, height, margin, padding)
- Proprietăți variate (cursor, pointer-events, content, animation, transition, transform)
- Proprietăți tipografice (font-size, font-family, text-align, text-transform)
- Proprietăți vizuale (color, background, border, box-shadow)

Posibile îmbunătățiri

Proiectul s-a bazat pe experiența proprie și pe postările studenților de pe Facebook. Pentru rezultate mult mai bune se pot efectua studii și statistici în vederea stabilirii principalelor probleme ale funcționalităților oferite în mod curent de către facultate, și frustrările și dorințele studenților legate de asemenea platforme educaționale.

Eta de *User Research* poate continua chiar dacă aplicația deja există: folosind chestionare elaborate și demo-uri sau folosind tool-uri special făcute pentru a verifica cât de frecventat, dezirabil și ușor de folosit este produsul dezvoltat. Candidați ce oferă aceste funcționalități ar putea fi: Google Analytics, Hotjar sau Mixpanel.

Aplicația curentă a fost testată pe Chrome și Chrome for Mobile, lăsând postprocesorul Autoprefixer să se ocupe de compatibilitatea între browser-e. Din păcate, acest lucru nu este de ajuns din cauza interpretărilor diferite ale proprietăților CSS. Așadar, o îmbunătățire majoră ar fi testarea și asigurarea acestei compatibilități pe browser-e precum Firefox, Internet Explorer, Edge, Safari și Safari Mobile.

În ceea ce privește latura *responsive*, au fost luate în considerare device-urile mobile și tabletele în modul portret. Deoarece unii studenții preferă să folosească tablete, mai ales în mod *landscape*, o îmbunătățire ar reprezenta optimizarea conținutului și pentru aceste dimensiuni de ecran.

O altă îmbunătățire ar fi un sistem de notificări asupra activității de pe platformă, astfel studentul ar ști exact când trebuie să intre pe site pentru a afla ce notă a luat sau ce curs a fost încărcat, etc.

Luând în considerare faptul că studenții întreabă mereu pe grupurile de Facebook cum poate fi contactat un anumit profesor, ar fi de mare ajutor o pagină special creată pentru a găsi datele acestora de contact.

Un alt lucru folositor ar fi posibilitatea de a viziona detalii (note, cursuri, etc.) pentru disciplinele deja promovate, lucru de care este nevoie când o disciplină curentă are la bază conceptele unei discipline deja studiate.

Pentru a percepe mai clar evenimentele de mare importanță, ar putea fi introdus un calendar. Acesta ar fi de mare ajutor pentru ilustrarea examenelor din timpul sesiunii.

Nu în ultimul rând, importantă ar fi o aplicație soră care ar fi implementată din punctul de vedere al profesorului, unde acesta ar putea încărca fișiere, adăuga note etc.

Concluzii

De-a lungul celor 3 ani de studiu am folosit aplicațiile universității (eSims), contul facultății (webmail), site-urile găzduite pe serverul facultății (paginile profesorilor) și de asemenea am urmărit discuțiile colegilor pe platforma de socializare Facebook. Cele mai des întâlnite întrebări au fost cele legate de autentificarea pe eSims, găsirea cursurilor, vizualizarea numărului curent de prezențe, a numărului minim de prezențe, găsirea parolelor aferente cursurilor, găsirea regulilor de promovare, găsirea site-ului unui profesor sau a unei pagini din acel site, găsirea orarului. De asemenea, nici unul din site-urile facultății nu sunt optimizate pentru telefoane sau tablete, device-uri foarte populare în zilele noastre.

Luând acestea în calcul, am creat o aplicație *proof of concept* ce răspunde unora dintre aceste întrebări, luând în seamă conceptele de UX, UI și UI development.

Prin aplicația FIIDashboard, am dorit să relievez importanța unui bun User Experience prin adaptarea unei interfețe la cunoștințele audienței țintă în scopul diminuării curbei de învățare (comportamentul elementelor din pagină este deja cunoscut datorită experiențelor acumulate de utilizator în experiențe anterioare, există indicatori pentru acțiunilor posibile asupra elementelor pentru a evita confuzia, se oferă feedback în urma acțiunilor utilizatorului).

Totodată, am ilustrat procesul de crearea a unui User Interface: wireframe-ul, mockup-ul iar prototipul a fost înlocuit cu o aplicație Angular deoarece aceasta a oferit mai multă libertate (în reprezentarea animațiilor, a modului responsive, etc.). Logo-ul, fonturile, culorile și spațierea reflectă identitatea aplicației, inspirând seriozitate, profesionalism și pot influența în mod subconștient starea de spirit a utilizatorului.

Pentru implementarea designului, m-am bazat pe funcționalitățile preprocesorului SCSS și pe postprocesoarele aferente Angular-CLI, am adăugat animații și am folosit media-queries pentru responsivitate.

Sunt de părere că o astfel de aplicație ar putea fi implementată în cadrul facultății prin implicarea studenților rezultând de asemenea și costuri mai mici. Menționez că pentru implementarea acestui tip de aplicație ar trebui efectuate inițial studii riguroase asupra nevoilor studenților și profesorilor. Activitatea s-ar desfășura la discipline diferite, provocarea fiind integrarea și asigurarea calității codului.

Bibliografie

- DeRome, J. (2015, August 13). *What is User Experience?* Preluat de pe User Testing Blog: <https://www.usertesting.com/blog/2015/08/13/what-is-user-experience/>
- Gallagher, N. (fără an). *normalize.css*. Preluat de pe Normalize.css: <https://necolas.github.io/normalize.css/>
- ISO/IEC JTC 1/SC 7. (2016, 06). *ISO/IEC 25066:2016(en) - Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for Usability — Evaluation Report*. Retrieved from ISO: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25066:ed-1:v1:en:term:3.23>
- Matyus, L. (fără an). *Improving HTML5 app performance with GPU accelerated CSS transitions*. Preluat de pe urbaninsight.com: <https://www.urbaninsight.com/article/improving-html5-app-performance-gpu-accelerated-css-transitions>
- Melbourne, B. (2011, 11 10). *The difference between a UX Designer and UI Developer*. Preluat de pe asinthecity.com: <https://asinthecity.com/2011/11/10/the-difference-between-a-ux-designer-and-ui-developer/>
- Morville, P. (2004, June 21). *User Experience Design*. Preluat de pe Semantic Studios: http://semanticstudios.com/user_experience_design/
- Norman, D. (2013). *The Design of Everyday Things*. New York: Basic Books.
- O'Connor, K. (2011, March 25). *Personas: The Foundation of a Great User Experience*. Preluat de pe UX Magazine: <https://uxmag.com/articles/personas-the-foundation-of-a-great-user-experience>
- Sass Basics*. (fără an). Preluat de pe sass-lang.com: <https://sass-lang.com/guide>
- Sass Documentation*. (fără an). Preluat de pe sass-lang.com: https://sass-lang.com/documentation/file.SASS_REFERENCE.html
- The Power of Sass and Why You Should Embrace CSS Preprocessors*. (2017, 02 28). Preluat de pe 1stwebdesigner.com: <https://1stwebdesigner.com/power-sass-why-use-css-preprocessors/>
- Tools, U. (2014). *How To Create UX Personas*. Preluat de pe Unamo: <https://unamo.com/blog/general/how-to-create-ux-personas>
- Usability.gov. (fără an). *Personas*. Preluat de pe Usability.gov: <https://www.usability.gov/how-to-and-tools/methods/personas.html>
- Usability.gov. (fără an). *User Interface Design Basics*. Preluat de pe Usability.gov: <https://www.usability.gov/what-and-why/user-interface-design.html>
- uxwriter.cc. (2017, May 28). *What is a UX Writer?* Preluat de pe <http://uxwriter.cc>: <http://uxwriter.cc/what-is-a-ux-writer/>
- Warcholinski, M. (2016, 04 20). *What Is The Difference Between Wireframe, Mockup and Prototype?* Preluat de pe brainhub.eu: <https://brainhub.eu/blog/difference-between-wireframe-mockup-prototype/>

