

# Puterea de REACTie

## Tema

Crearea de componente adaptive pentru o aplicație web.

## De ce ați ale acest proiect?

### Motivația

Am ales această temă propusă de către Nokia, deoarece reprezintă o soluție a problemei pe care am avut-o, în semestrul doi al primului an de facultate, în dezvoltarea proiectului de la materia Elemente de Web Design și anume crearea unui site adaptiv.

De asemenea, această experiență mă va ajuta să îmi dezvolt cunoștințele de HTML, CSS, JS (React) și nu numai, va fi un plus pentru viitoarea mea carieră, dar și în dezvoltarea personală.

Consider că pentru un viitor web developer este important să știe să își construiască produsul astfel în cât să se plieze cât mai bine pe dorințele și modul de viață al utilizatorului. Având în vedere că astăzi paginile web sunt afișate pe mai multe tipuri de dispozitive, cum ar fi tablete, smartphone-uri, cărți electronice și calculatoare, chiar mai mult, fiecare dispozitiv are caracteristicile sale specifice: dimensiunea ecranului, rezoluție, putere CPU și capacitatea de memorie. Dezvoltarea unui site web adaptiv este fundamental în creșterea numărului de utilizatori.

### Actualitate temă

React este cea mai importantă bibliotecă pentru dezvoltare de UI, fiind un instrument foarte practic ce permite programatorilor să dezvolte o interfață ușor de utilizat în cel mai scurt timp. ReactJS ne permite să creăm repede un frontend scalabil și ușor de utilizat pentru aplicațiile web. Este unul dintre opțiunile open-source cele mai populare în rândul programatorilor dar și al mediului de afaceri, mulțumită avantajelor sale în dezvoltarea aplicațiilor web: permite dezvoltarea de aplicații la scară largă de tip SPA (single page application), în scurt timp și la un nivel superior de calitate, permite scrierea de cod curat, modular și reutilizabil, este ideal pentru proiecte ce pot fi sparte în componente separate, aplicațiile web construite în ReactJS sunt flexibile, se pliază pe cerințele SEO și sunt ușor de scalat și de menținut, este ușor să se treacă la React Native și să se creeze aplicații mobile cu un aspect nativ.

### Domeniul de aplicabilitate

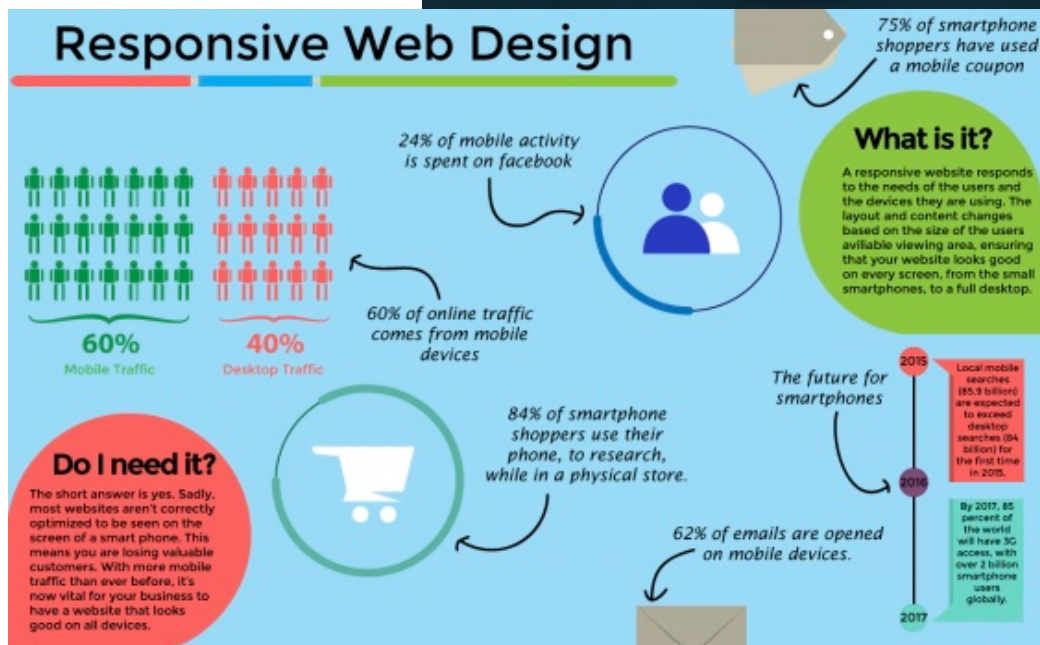
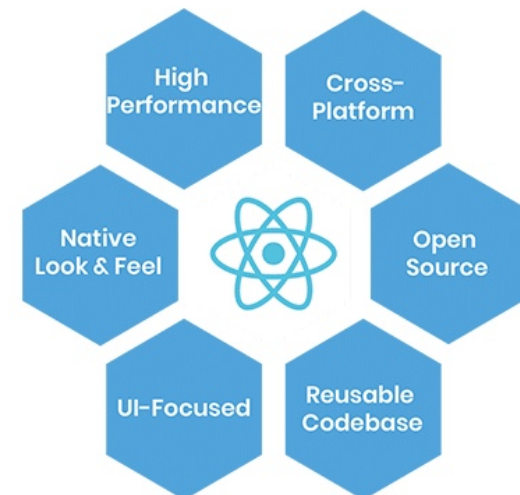
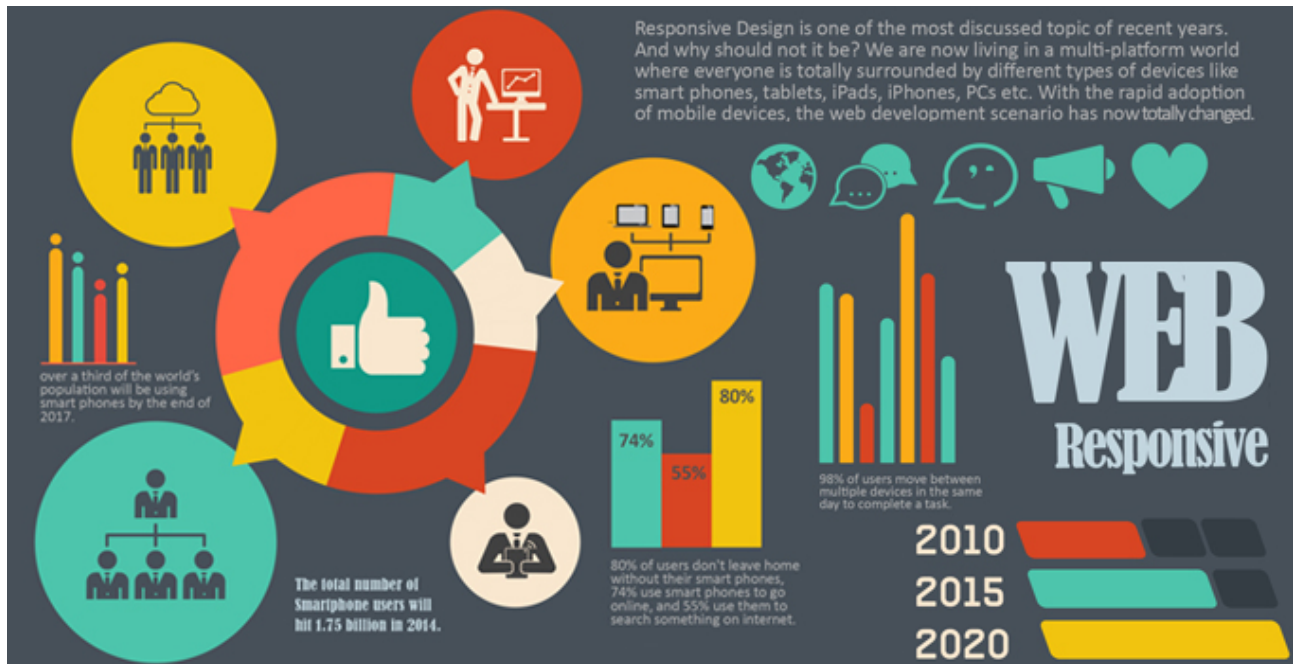
Tema aleasă afectează mai multe componente ale unei aplicații web ce are ca și caracteristici adaptabilitatea la diverse tipuri de dispozitive. Aceasta face parte, de asemenea, din procesul de dezvoltare de aplicații web, proces ce se desfășoară de la momentul conceperii structurii și interfeței grafice și până la finalizarea programării și introducerea propriu-zisă a datelor care alcătuiesc conținutul site-ului.

### Potențialii utilizatori

Fiind o componentă a aplicației web “Talent Management Platform”, proiectul va fi vizibil pentru toți utilizatorii ce accesează site-ul. Interacțiunea cu ceea ce am lucrat eu va fi implicită la accesare aplicația web având în vedere tematica, faptul că ar trebui să se adapteze la dimensiunile și specificul dispozitivului de pe care este accesată.



## De ce responsive?



## Obiectivul

Obiectivul proiectului este de a mă folosi de beneficiile aduse de CSS media queries și flexbox în dezvoltarea unui design web adaptiv, pentru a schimba randarea a header-ului, footer-ului și sidebar-ului în funcție de dispozitivul folosit în accesarea site-ului web. Astfel facilitând experiența utilizatorului, pentru că viitorul web design-ului nu mai este despre ceea ce putem face, ci mai degrabă despre ceea ce ar trebui să facem. Asta înseamnă să fii atent la modul în care designul poate afecta persoanele care îl folosesc și la proiectarea site-urilor web care au ca rezultat experiențe pozitive pentru utilizatori.

## Descrierea proiectului

Pentru a-mi îndeplini obiectivul voi colabora cu coordonatorul proiectului Nokia pentru care am fost acceptată, iar în primele două săptămâni mă voi familiariza cu librăria React Java Script și îmi voi dezvolta aptitudinile deja dobândite. După cele două săptămâni de antrenament mă voi strădui să dezvolt o versiune beta care până la finalul semestrului ar trebui să reprezinte rezultatul final.

În acest proiect, de-a lungul perioadei menționate voi folosi editorul Visual Studio Code, dar și GIT ca SCM (Source Code Manager) în dezvoltarea proiectului. Librăria React va sta la baza proiectului ceea ce semnifică faptul că pentru a începe să creez o aplicație cu React voi avea nevoie și de Node.js pentru execuție. În procesul de management al pachetelor din Node.js JavaScript va fi antrenat implicit și software-ul npm. Toate pachetele npm sunt definite în fișiere numite package.json, npm poate fi instalat având toate modulele sau având doar modulele necesare proiectului. În realizarea proiectului vor fi necesare două module în special @material-ui/lab, @material-ui/core, instalate prin intermediul npm, dar și React Router ce va ajuta la navigarea între vizualizările diferitelor componente ale aplicației React, permițând modificarea adresei URL a browserului și menținerea interfeței de utilizare sincronizată cu adresa URL. Bineînțeles că în procesul de creație vor fi necesare și noțiuni privind UI, astfel încât interfața aplicației să fie atractivă, stimulativă vizual și tematic adecvată pentru a se potrivi cu scopul și personalitatea aplicației.

## Conceptul introdus

Conceptul în jurul căruia gravitează proiectul ales și anume responsive web design, îi aparține lui Ethan Marcotte care în 2010 a introdus o modalitate diferită de a utiliza HTML și CSS. Ideea principală care a stat la baza proiectării responsive a fost aceea că un singur site web ar putea răspunde și s-ar putea adapta la diferite medii de afișare, facilitând utilizarea pe diferite dispozitive.

## Arhitectura proiectului

Momentan aplicația are o structură de tip monolit având un singur nivel în care interfața utilizatorului și codul de acces la date sunt combinate într-un singur program. Investigăm beneficiile migrării la una de tip microservicii astfel dezvoltând o arhitectură Micro-Frontend. Arhitectura Micro-Frontend presupune ca fiecare componentă să fie o aplicație React completă. Toate acestea sunt compuse într-un singur fișier index.html furnizat de hub. Hub-ul gestionează, de asemenea, rutarea la nivel de rădăcină pentru a determina ce componentă să încarce. Spre deosebire de alte librării și cadre UI, ReactJs nu impune un model de arhitectură. Este doar o vizualizare care se adresează interfeței utilizatorului. O componentă React nu este altceva decât unitatea structurală centrală - ceva minimalist, cum ar fi un buton, o etichetă sau o etichetă de introducere a textului sau poate ceva mai complex, cum ar fi un formular de înregistrare, un profil de utilizator etc. Toate componentele din React pot avea o stare. Prin stare, ne referim la datele necesare în monitorizarea funcționalității aplicației. De asemenea, starea aplicației se schimbă din când în când, în funcție de acțiunea întreprinsă de utilizator. Doar starea datelor determină ce se afișează în interfața de utilizare a unei aplicații.



## Indicatorii de performanță

Când site-ul se încarcă pe un dispozitiv, acesta ar trebui să-și schimbe forma pentru a se potrivi dispozitivului respectiv. Dacă încă arată ca pe un desktop, dar este doar micșorat pe ecranul dispozitivului, experiența utilizatorului nu va fi cea mai prietenoasă.

## Tehnologiile utilizate

- I. **React** este o librărie JavaScript open-source pentru construirea de interfețe de utilizator de tip SPA. React a fost conceput pentru o adoptare treptată, astfel se poate folosi indiferent dacă se dorește adăugarea unor elemente interactive într-o pagină HTML simplă sau începerea unei aplicații complexe bazată pe React.
- II. **Node.js** este un mediu de execuție JavaScript de tip back-end open-source, cross-platform, care rulează pe motorul V8 și execută cod JavaScript în afara unui browser web. Node.js oferă, de asemenea, o bibliotecă bogată cu diverse module JavaScript care simplifică dezvoltarea aplicațiilor web folosind Node.js.
- III. **npm** (node package manager) este managerul de pachete pentru platforma JavaScript Node. Pune modulele la locul lor, astfel încât nod să le poată găsi și să gestioneze în mod inteligent conflictele de dependență. Cel mai frecvent, este folosit pentru a publica, descoperi, instala și dezvolta programe nod.
- IV. **JavaScript** este un limbaj de programare orientat obiect bazat pe conceptul prototipurilor. Este folosit mai ales pentru introducerea unor funcționalități în paginile web, codul JavaScript din aceste pagini fiind rulat de către browser. Limbajul este binecunoscut pentru folosirea sa în construirea siturilor web, dar este folosit și pentru accesul la obiecte încapsulate (embedded objects) în alte aplicații.
- V. **CSS** sau Cascading Style Sheets este un standard pentru formatarea elementelor unui document HTML. Este limbajul folosit pentru a descrie felul în care va arăta o pagină Web, inclusiv culorile, aspectul și fonturile.  
**Media queries** este o caracteristică a CSS 3 care permite conținutului să se adapteze la diferite tipuri de dispozitive. Pot fi folosite pentru a verifica multe lucruri, cum ar fi: lățimea și înălțimea ferestrei de vizualizare, lățimea și înălțimea dispozitivului, orientare (tableta/telefonul este în modul orizontal sau portret) și rezoluție.  
**Flexbox** Layout își propune să ofere o modalitate mai eficientă de a aranja, alinia și distribui spațiul între articolele dintr-o casetă, chiar și atunci când dimensiunea acestora este necunoscută și/sau dinamică. Ideea principală din spatele aspectului flexibil este de a oferi casetei capacitatea de a modifica lățimea/înălțimea articolelor pentru a umple cât mai bine spațiul disponibil (mai ales pentru a se adapta la toate tipurile de dispozitive).
- VI. **HTML** ( Hypertext Markup Language ) este unul din primele elemente fundamentale ale WWW ( World Wide Web ), care descrie formatul primar în care documentele sunt distribuite și văzute pe Web. Multe din trasaturile lui, cum ar fi independența față de platforma, structurarea formătărilor și legăturile hypertext, fac din el un foarte bun format pentru documentele Internet și Web.

## Baze de date

În proiect nu voi lucra cu baze de date, deoarece mă voi axa pe elemente ale aplicației client și nu asupra componentelor backend: header, footer, sidebar adaptive astfel în cât să aibă capacitatea să-și adapteze designul paginilor în funcție de mediul de vizionare, fără a suferi degradări majore, pentru a le oferi utilizatorilor o experiență excelentă de navigare. În tot acest proces ne fiind nevoie să stocăm date.





## Terminologie

**UI** (interfață de utilizator) este aspectul grafic al unei aplicații. Acesta constă din butoanele pe care utilizatorii dau clic, textul pe care îl citesc, imagini, glisoare, câmpuri de introducere a textului și toate celelalte elemente cu care interacționează utilizatorul. Aceasta include aspectul ecranului, tranzițiile, animațiile de interfață și fiecare micro-interacțiune. Orice fel de element vizual, interacțiune sau animație trebuie să fie proiectată.

**UX** (experiența utilizatorului) se concentrează pe studierea modului în care utilizatorii se deplasează prin toate elementele diferite ale interfeței UI. Pentru designerii UX, este vorba despre optimizarea fluxului experienței utilizatorului și eliminarea oricăror surse de probleme în timpul acestor interacțiuni. Un designer UX este responsabil pentru ca produsele să fie utile, plăcute și accesibile utilizatorilor.



## Codul Sursă

```
->SRC
  ->COMPONENTS
    ->COMMON
      ->LoginError
        ->LoginError.scss
        ->LoginError.tsx
      ->Create.tsx
      ->ExportExcel.tsx
      ->Form.scss
      ->Form.tsx
      ->Modal.tsx
    ->Credits
      ->Credits.css
      ->Credits.tsx
      ->jCredist.json
    ->Footer
      ->Footer.scss
      ->Footer.tsx
    ->Header
      ->Header.scss
      ->Header.tsx
    ->Homepage
      ->Homepage.scss
      ->Homepage.tsx
      ->sun.scss
    ->LandingPage
      ->LandingPage.scss
      ->LandingPage.tsx
    ->Newsletter
      ->newsletter.scss
      ->Newsletter.tsx
      ->jNewsletter.json
      ->FirstArticol
        ->FirstArticol.tsx
        ->FirstArticol.scss
        ->jFirstArticol1.json
      ->SecontArticol
        ->SecontArticol.tsx
        ->SecontArticol.scss
        ->jSecontArticol1.json
      ->ThirdArticol
        ->ThirdArticol.tsx
        ->ThirdArticol.scss
        ->jThirdArticol1.json
    ->SideBar
      ->SideBar.scss
      ->SideBar.tsx
      ->SideBarData.js
      ->SubMenu.tsx
  ->toggleSwitch
```



- >ToggleSwitch.scss
  - >ToggleSwitch.tsx
- >LanguageSelector.tsx
- >UseMemo.tsx
- >designSystems
  - >genericModal.scss
  - >genericModal.tsx
- >HOCs
  - >authGuard.js
- >hooks
  - >utils.tsx
- >images
- >middlewares
  - >LogOut.tsx
- >reducers
  - >auth.js
  - >index.js
- >redux
  - >actions
    - >projects
      - >enableButton.js
      - >getProjects.js
      - >reviewTask.js
      - >setDefaultValue.js
      - >setTaskCompetenceLevel.js
      - >toggleDisplay.js
    - >tasks
      - >getTasks.js
      - >onDeleteTask.js
      - >setTask.js
  - >reducers
    - >auth.js
    - >counter.js
    - >index.js
    - >islogged.js
    - >projects.js
    - >tasks.js
    - >types.js
- >services
  - >i18n.js
- >App.js
- >App.test.js
- >AppRouter.js
- >config.js
- >index.css
- >index.tsx
- >main.scss
- >registerServiceWorker.ts



### LoginError.scss

```
h3 {  
  padding: 250px 0;  
  text-align: center;  
}
```

### LoginError.tsx

```
import React from "react";  
import "../LoginError.scss"  
import { withTranslation } from "react-i18next";  
import { useTranslation } from "react-i18next";  
const LoginError = () => {  
  return(<h3>Please contact <a  
href="mailto:alexandru.bran@nokia.com?subject=support  
request&body=Please provide access to the following apps in the  
azure platform:">alexandru.bran@nokia.com</a> for support</h3>)  
}  
export default LoginError
```

### Create.tsx

```
import { useState } from "react";  
import FormPropsTextFields from "../Form";  
import regeneratorRuntime from "regenerator-runtime";  
const initialFormValues = {  
  appel: '',  
  data: '',  
  data2: '',  
  abran: '',  
  target: '',  
  ericson: '',  
  ttType: '',  
  flag: '',  
  upalu: '',  
  language: '',  
  si: '',  
  problematique: '',  
  tt: '',  
  itv: '',  
  onSait: '',  
  norm: '',  
  mainCause: '',  
  action: '',  
  ttCreator: '',  
  technician: '',  
  collage: '',  
  time: '',  
  sousCause: '',  
}
```





```

corectiveAction: '',
  processStatus: '',
  inverted: '',
  bagot: '',
  active: '',
  formSubmitted: false,
  success: false,
  id: '',
}

export const useFormControls = () => {
  const [values, setValues] = useState(initialFormValues);
  const [errors, setErrors] = useState({} as any);
  const validate: any = (fieldValues = values) => {
    let temp: any = { ...errors }
    if ("appel" in fieldValues)
      temp.appel = fieldValues.appel ? "" : "This field is
required."
    if ("flag" in fieldValues)
      temp.flag = fieldValues.flag ? "" : "This field is
required."
    if ("tt" in fieldValues)
      temp.tt = fieldValues.tt ? "" : "This field is required."
    if ("ttCreator" in fieldValues)
      temp.ttCreator = fieldValues.ttCreator ? "" : "This field
is required."
    if ("data" in fieldValues)
      temp.data = fieldValues.data ? "" : "This field is
required."

    if ("data2" in fieldValues)
      temp.data2 = fieldValues.data2 ? "" : "This field is
required."
    if ("time" in fieldValues)
      temp.time = fieldValues.time ? "" : "This field is
required."
    if ("itv" in fieldValues)
      temp.itv = fieldValues.itv ? "" : "This field is
required."
    if ("technician" in fieldValues)
      temp.technician = fieldValues.technician ? "" : "This
field is required."
    if ("abran" in fieldValues)
      temp.abran = fieldValues.abran ? "" : "This field is
required."
    if ("upalu" in fieldValues)
      temp.upalu = fieldValues.upalu ? "" : "This field is
required."
    if ("onSait" in fieldValues)
      temp.onSait = fieldValues.onSait ? "" : "This field is
required."

```



```

        if ("collage" in fieldValues)
            temp.collage = fieldValues.collage ? "" : "This field is
required."

        if ("target" in fieldValues)
            temp.target = fieldValues.target ? "" : "This field is
required."

        if ("language" in fieldValues)
            temp.language = fieldValues.language ? "" : "This field is
required."
        if ("norm" in fieldValues)
            temp.norm = fieldValues.norm ? "" : "This field is
required."
        if ("ericson" in fieldValues)
            temp.ericson = fieldValues.ericson ? "" : "This field is
required."
        if ("si" in fieldValues)
            temp.si = fieldValues.si ? "" : "This field is required."
        if ("mainCause" in fieldValues)
            temp.mainCause = fieldValues.mainCause ? "" : "This field
is required."
        if ("sousCause" in fieldValues)
            temp.sousCause = fieldValues.sousCause ? "" : "This field
is required."
        if ("ttType" in fieldValues)
            temp.ttType = fieldValues.ttType ? "" : "This field is
required."
        if ("problematique" in fieldValues)
            temp.problematique = fieldValues.problematique ? "" :
"This field is required."
        if ("action" in fieldValues)
            temp.action = fieldValues.action ? "" : "This field is
required."
        if ("corectiveAction" in fieldValues)
            temp.corectiveAction = fieldValues.corectiveAction ? "" :
"This field is required."
        if ("processStatus" in fieldValues)
            temp.processStatus = fieldValues.processStatus ? "" :
"This field is required."
        if ("inverted" in fieldValues)
            temp.inverted = fieldValues.inverted ? "" : "This field is
required."
        if ("bagot" in fieldValues)
            temp.bagot = fieldValues.bagot ? "" : "This field is
required."
        if ("active" in fieldValues)
            temp.active = fieldValues.active ? "" : "This field is
required."
        setErrors({
            ...temp
        });

```



```

    }
    const handleInputValue = (e: any, name:string) => {
      setValues({
        ...values,
        [name]: e.target.value
      });
      validate({ [name]: e.target.value });
      console.log(name);
      console.log(e.target.value);
      console.log(values);
    };
    const handleFormSubmit = async (e: any) => {
      e.preventDefault();
      await FormPropsTextFields(values);
      alert(JSON.stringify(e));
    };
    const formIsValid = (fieldValues = values) => {
      const isValid =
        fieldValues.appel &&
        fieldValues.abran &&
        fieldValues.flag &&
        fieldValues.tt &&
        fieldValues.ttCreator &&
        Object.values(errors).every((x) => x === "");
      return isValid;
    };
    return {
      handleInputValue,
      handleFormSubmit,
      formIsValid,
      errors
    };
  }
}

```

### **ExportExcel.tsx**

```

import React from 'react'
import * as FileSaver from "file-saver";
import * as XLSX from "xlsx";
import Button from '@material-ui/core/Button';
export const ExportToExcel = ({apiData, fileName, operationName})
=> {
  const fileType =
    "application/vnd.openxmlformats-
officedocument.spreadsheetml.sheet;charset=UTF-8";
  const fileExtension = ".xlsx";

  const exportToCSV = (apiData,fileName) => {
    const ws = XLSX.utils.json_to_sheet(apiData);
    const wb = { Sheets: { data: ws }, SheetNames: ["data"] };
    const
      excelBuffer =

```



```

XLSX.write(wb, { bookType: "xlsx", type: "array" });
  const data = new Blob([excelBuffer], { type: fileType });
  FileSaver.saveAs(data, fileName + fileExtension);
};
return (
  <Button variant="contained" color="primary" onClick={(e) =>
exportToCSV(apiData, fileName)}>{operationName}</Button>
  );
};

```

### **Form.scss**

```

.add-form {
  text-align: center;
  border: 2px solid red;
  display: flex;
  justify-content: center;
  margin-top: 50px;
}

```

### **Form.tsx**

```

import React from 'react';
import { createStyles, makeStyles, Theme } from '@material-ui/
core/styles';
import { useForm, Controller } from 'react-hook-form'
import Button from '@material-ui/core/Button';
import Autocomplete from '@material-ui/lab/Autocomplete';
import TextField from '@material-ui/core/TextField';
import MenuItem from '@material-ui/core/MenuItem';
import { InputAdornment } from '@material-ui/core';
import Grid from "@material-ui/core/Grid";
import Typography from "@material-ui/core/Typography";
import Divider from "@material-ui/core/Divider";
import { useFormControls } from './Create';
import './Form.scss'
const projectTypes = [
  {
    value: 'T2',
    label: 'Summer Practice',
  },
  {
    value: 'T1',
    label: 'Individual Project',
  },
  {
    value: 'T3',
    label: 'Team Project',
  },
  {
    value: 'T4',

```



```

    label: 'Single Day Event',
  },
];
const projectTypes2 = ['Summer Practice', 'Individual Project',
'Team Project', 'Single Day Event']
const useStyles = makeStyles((theme: Theme) =>
  createStyles({
    root: {
      '& .MuiTextField-root': {
        margin: theme.spacing(1),
        width: '25ch',
      },
    },
    primaryColor: {
      color: 'rgb(169,169,169)',
    },
    secondaryColor: {
      color: 'blue'
    },
    label: {
      backgroundColor: 'rgb(169,169,169)',
      width: 'auto',
      height: '56px',
      border: '1px groove black',
      display: 'flex',
      alignItems: 'center',
    },
    right: {
      float: 'right',
      display: 'flex',
      flexDirection: 'column',
      marginRight: '150px',
    },
  })),
);
type Profile = {
  appel: string
  data: string
  data2: string
  abran: string
  target: string
  ericson: string
  ttType: string
  flag: string
  upalu: string
  language: string
  si: string
  problematique: string
  tt: string
  itv: string
  onSait: string
  norm: string

```





```

mainCause: string
action: string
ttCreator: string
technician: string
collage: string
time: string
sousCause: string
corectiveAction: string
processStatus: string
inverted: string
bagot: string
active: string
}
export default function FormPropsTextFields(props: any) {
  const classes = useStyles();
  const { register, handleSubmit, control } = useForm({});
  const {
    handleInputValue,
    handleFormSubmit,
    // formIsValid,
    errors
  } = useFormControls();
  console.log(props)
  const onSubmit = (data: any) => { console.log({props});
props.saveFunction(data) }
  return (
    <Grid container>
      <div className="add-from">
        <form
          className={classes.root}
          autoComplete="off"
          onSubmit={handleSubmit(onSubmit)}
        >
          <Grid item xs={12}>
            <Typography className={classes.primaryColor} >
              Main Info
            </Typography>
            <Divider
              style={{ marginTop: 20 }}
            />
            <Grid container direction="row"
              className={classes.mainHeader}>
              <Controller
                name="id"
                control={control}
                defaultValue={props.operation === 'edit' ?
props.values.id : null}
                render={({ field: { onChange, value }, fieldState:
{ error } }) => (
                  <TextField
                    id="id"

```



```

        type="text"
        label="Project id"
        className={classes.textField}
        onChange={onChange}
        disabled={true}
        error={!!error}
        helperText={error ? error.message : null}
        InputLabelProps={{
          shrink: true,
        }}
        {...register('id')}
      />
    )}
  />
  <Controller
    name="title"
    control={control}
    render={({ field: { onChange, value }, fieldState:
{ error } }) => (
      <TextField
        id="title"
        type="text"
        label="Project title"
        className={classes.textField}
        defaultValue={props.operation === 'edit' ?
props.values.title : null}
        onChange={onChange}
        error={!!error}
        helperText={error ? error.message : null}
        InputLabelProps={{
          shrink: true,
        }}
        {...register('title')}
      />
    )}
    rules={{ required: 'Project Title is required' }}
  />
  <Controller
    control={control}
    name="type"
    defaultValue={props.operation === 'edit' ?
props.values.type : null}
    render={({ field: { onChange, value }, fieldState:
{ error } }) => (
      <Autocomplete
        value={value}
        onChange={(event, item) => {
          onChange(item);
        }}
        id="type"
        options={projectTypes2}

```



```

        renderInput=({ params } => (
          <TextField
            {...params}
            error={!error}
            helperText={error ? error.message : null}
            label="type"
          />
        ))
      />
    )}
    rules={{ required: 'Project type is required' }}
  />
  <Controller
    name="coordinator"
    control={control}
    render={({ field: { onChange, value }, fieldState:
{ error } }) => (
      <TextField
        id="coordinator"
        type="text"
        label="Project coordinator"
        className={classes.textField}
        defaultValue={props.operation === 'edit' ?
props.values.coordinator : null}
        onChange={onChange}
        error={!error}
        helperText={error ? error.message : null}
        InputLabelProps={{
          shrink: true,
        }}
        {...register('coordinator')}
      />
    )}
    rules={{ required: 'Project coordinator is
required' }}
  />
</Grid>
</Grid>
<Grid container direction="row"
className={classes.mainHeader}>
  <Grid item xs={12}>
    <Typography className={classes.primaryColor} >
      Details
    </Typography>
    <Divider
      style={{ marginTop: 20 }}
    />
    <Controller
      name="description"
      control={control}
      render={({ field: { onChange, value }, fieldState:
{ error } }) => (

```



```

        <TextField
          id="description"
          type="text"
          label="Project description"
          className={classes.textField}
          defaultValue={props.operation === 'edit' ?
props.values.description : null}
          onChange={onChange}
          multiline
          rows={14}
          error={!error}
          helperText={error ? error.message : null}
          InputLabelProps={{
            shrink: true,
          }}
          {...register('description')}
        />
      )}
      rules={{ required: 'Project description is
required' }}
    />
    <Controller
      name="requirements"
      control={control}
      render={({ field: { onChange, value }, fieldState:
{ error } }) => (
        <TextField
          id="requirements"
          type="text"
          label="Project requirements"
          defaultValue={props.operation === 'edit' ?
props.values.requirements : null}
          className={classes.textField}
          onChange={onChange}
          multiline
          rows={14}
          error={!error}
          helperText={error ? error.message : null}
          InputLabelProps={{
            shrink: true,
          }}
          {...register('requirements')}
        />
      )}
      rules={{ required: 'Project requirements are
required' }}
    />
  </Grid>
</Grid>
  <Button type="submit" variant="contained"
color="primary">
    Add

```



```

        </Button>
      </form>
    </div>
  </Grid>
);}

```

### Modal.tsx

```

import React from 'react';
import { makeStyles, Theme, createStyles } from '@material-ui/
core/styles';
import Modal from '@material-ui/core/Modal';
import Button from '@material-ui/core/Button';
import FormPropsTextFields from './Form';
import CloseIcon from '@material-ui/icons/Close';
import DoneIcon from '@material-ui/icons/Done';
function getModalStyle() {
  return {
    width: '100%',
    maxWidth: '100vw',
    maxHeight: '100%',
    position: 'fixed',
    top: '50%',
    left: '0%',
    transform: 'translate(0, -50%)',
    overflowY: 'auto'
  };
};
}
const useStyles = makeStyles((theme: Theme) =>
  createStyles({
    paper: {
      position: 'absolute',
      width: 1500,
      backgroundColor: theme.palette.background.paper,
      border: '2px solid #000',
      boxShadow: theme.shadows[5],
      padding: theme.spacing(2, 4, 3),
    },
    button: {
      float: 'right',
      backgroundColor: 'red',
    },
  })),
);
export default function SimpleModal(props:any) {
  const classes = useStyles();
  const [modalStyle] = React.useState(getModalStyle);
  console.log(props)
  const body = (
    <div style={modalStyle} className={classes.paper}>
      <div><b>{props.title}</b></div>

```





```

    <button className={classes.button} type="button"
onClick={props.handleModal}>
      <CloseIcon />
    </button>
    <FormPropsTextFields
values={props.item.data}
operation={props.operation}
handleInputValues={props.handleInputValues}
saveFunction={props.saveFunction}
/>

  </div>
);
return (
  <div>
    <Modal
      open={props.handleModal}
      onClose={props.handleClose}
      aria-labelledby="simple-modal-title"
      aria-describedby="simple-modal-description"
    >
      {body}
    </Modal>
  </div>

);
}

```

### Credits.css

```

@media (max-width:1000px) {
  :root {
    --top: 700px
  }
}
@media (min-width:1000px) {
  :root {
    --top:900px
  }
}
.credits-container {
  width: 100%;
  height: 100%;
  margin: 0;
  min-height: 100vh;
  background: #a2d4ff;
  background-attachment: fixed;
  background-image:
    radial-gradient(rgba(19, 170, 216), rgba(57, 86, 250, 0.767)
2px, transparent 40px),

```



```

        radial-gradient(rgb(19, 170, 216), rgba(71, 124, 238, 0.904)
1px, transparent 30px),
        radial-gradient(rgb(19, 170, 216), rgba(77, 195, 250, 0.849)
2px, transparent 40px),
        radial-gradient(rgba(19, 170, 216,.4), rgba(25, 102, 202,
0.726) 2px, transparent 30px),
        radial-gradient(rgba(19, 170, 216,.4), rgba(36, 141, 211,
0.726) 2px, transparent 30px),
        radial-gradient(#3049d6, #0820ac,#06156e ,#040d42,#02092e);
        background-size: 350px 350px, 300px 300px, 250px 250px,
200px 200px, 160px 160px,cover;
        background-position: center;
        overflow: hidden;
        font-family: "Lucida Console", Courier , monospace;
        position: fixed;
    }
.credits-container::after {
    content: '';
    position: fixed;
    top:0px;
    width:100%;
    height: 60%;
    background: linear-gradient(180deg, rgba(62, 71, 201, 0.993)
40%, rgba(67, 77, 216, 0) 100%)
}
.wrapper {
    display: flex;
    height: 100%;
    width: 80%;
    margin-left: 10%;
    perspective: 1300px;
}
.scroll-text {
    color:#ffffff;
    font-size:2vw;
    line-height:1.53em;
    font-weight: bold;
    text-align:justify;
    position: relative;
    animation: scroll 40000s linear forwards;
}
h1, h2 {
    text-align: center;
}@keyframes scroll {
    from {top: var(--top); transform: translateZ(0) rotateX(0deg)}
    to {top: -7550000px; transform: translateZ(-2500px)
rotateX(5deg) }
}.image{
    width:15vw;
    height:15vw;
    float: left;
    margin: 0 15px 0
0;

```



```

}
.nume, .ocupatie, .descriere {
  size:10vw;
}

```

### Credits.tsx

```

import React from "react"
import "./Credits.css"
import { config } from "../../config";
import ReactAudioPlayer from "react-audio-player";
import CardDeck from "react-bootstrap/CardDeck";
import Card from "react-bootstrap/Card";
import Container from "react-bootstrap/Container";
import Row from "react-bootstrap/Row";
import {Component} from "react";
import data from "../jCredist.json";
export const Credits = () => {
  const totalItems = 100;
  const items = new Array(totalItems).fill(null);
  const url = config.baseURL + "/158066369-space-wars.m4a";
  return(
    <div className="credits-container" >
      <div className="wrapper">
        <div className="scroll-text" >
          <Container fluid={true}>
            <Row>
              {items.map((_, idx) =>
                <CardDeck className=" no-gutters " key = {idx}>
                  {data.map((postData) => {console.log(postData);
                    return (
                      <div className="credit-persoana">
                        <img src={postData.image className="image"/>
                        <div className="tabel">
                          <div className="nume">
                            Nume:{postData.nume}
                          </div>
                          <div className="ocupatie">
Ocupatie:{postData.ocupatie}
                          </div>
                          </div>
                          <br></br>
                        <div className="descriere">{postData.descriere}</div>
                          <br></br>
                        </div>
                      </div>
                    )};
                  </div>
                </div>
              )}
            </CardDeck>
          </Row>
        </Container>
      </div>
    </div>
  );
}

```



```

</div>
</div></div>
)
}

```

### jCredits.json

```

[
  {
    "userId": 1,
    "id": 1,
    "nume": " Nume1",
    "ocupatie": "Ocupatie1",
    "descriere": " Because he takes nsuscipit accepted result
lightly with nreprehenderit discomfort may be the entire
nnostrum of the things that happens is that they are extremely ",
    "image": "https://source.unsplash.com/rDEOVtE7vOs/
1600x900"
  },
  {
    "userId": 1,
    "id": 2,
    "nume": " Nume2",
    "ocupatie": "Ocupatie2",
    "descriere": " Because he takes nsuscipit accepted result
lightly with nreprehenderit discomfort may be the entire
nnostrum of the things that happens is that they are extremely ",
    "image": "https://source.unsplash.com/
WNoLnJo7tS8/1600x900"
  },
  {
    "userId": 1,
    "id": 3,
    "nume": " Nume3",
    "ocupatie": "Ocupatie3",
    "descriere": " Because he takes nsuscipit accepted result
lightly with nreprehenderit discomfort may be the entire
nnostrum of the things that happens is that they are extremely ",
    "image": "https://source.unsplash.com/
vMV6r4VRhJ8/1600x900"
  },
  {
    "userId": 1,
    "id": 4,
    "nume": " Nume4",
    "ocupatie": "Ocupatie4",
    "descriere": " Because he takes nsuscipit accepted result
lightly with nreprehenderit discomfort may be the entire
nnostrum of the things that happens is that they are extremely ",
    "image": "https://source.unsplash.com/mEZ3PoFGs_k/
1600x900"
  },

```



```

    {
      "userId": 1,
      "id": 5,
      "nume": " Nume5",
      "ocupatie": "Ocupatie5",
      "descriere": " Because he takes nsuscipit accepted result
lightly with nreprehenderit discomfort may be the entire
nnostrum of the things that happens is that they are extremely ",
      "image": "https://source.unsplash.com/6W4F62sN_yI/
1600x900"
    }
  ]

```

### **Footer.scss**

```

* {
  box-sizing: border-box;
  font-family: "Lato", sans-serif;
  margin: 0;
  padding: 0;
}
ul {
  list-style: none;
}
.ft-main ul {
  margin: 0;
  padding: 0;
  text-align: center;
  margin-left: -4%;
}
.footer {
  display: flex;
  justify-content: center;
  overflow: hidden;
  background-color: #2196F3;
  bottom: 0;
  left: 0;
  width: 100%;
  z-index: 100;
}
.footer a {
  text-decoration: none;
  color: #eee;
  text-align: center;
}
a:hover {
  text-decoration: bold;
}
.ft-title {

```





```

    color: #fff;
    font-family: "Merriweather", serif;
    font-size: 1.375rem;
    text-align: center;
  }
body {
  display: flex;
  min-height: 100vh;
  flex-direction: column;
}
.container {
  flex: 1;
}
.ft-main {
  display: flex;
  flex-wrap: wrap;
}
@media only screen and (max-width: 400px) {
  .ft-main-item{
    margin: 0 auto;
    text-align: center;
    width: 10%;
  }
  .ft-main-item form{
    flex-wrap: nowrap;
    justify-content: center;
  }
  .ft-main {
    flex-wrap: wrap;
    justify-content: space-evenly;
  }
}

@media only screen and (min-width:400px) and (max-width:600px){
  .ft-main-item{
    margin: 0 auto;
    text-align: center;
    width: 10%;
  }
  .ft-main-item form{
    flex-wrap: nowrap;
    justify-content: center;
  }
  .ft-main {
    justify-content: space-evenly;
    flex-direction: column;
  }
}

@media only screen and (min-width: 600px){
  .ft-main {
    justify-          content: space-

```



```

evenly;
    }
    .footer {
        position: absolute;
    }
}
.ft-main-item1{
    padding-right: 2.50rem
}
.ft-main-item-234 {
    color: #fff;
    min-width: 12.5rem;
}
form {
    display: flex;
    color: #fff;
    flex-wrap: wrap;
}
input[type="email"] {
    border: 0;
    padding: 0.625rem;
    width: 16rem;
    margin-bottom: 0.3125rem;
}
input[type="submit"] {
    background-color: #00d188;
    color: #fff;
    cursor: pointer;
    border: 0;
    padding: 0.625rem 0.9375rem;
    margin-bottom: 0.3125rem;
}
.ft-social-list {
    display: flex;
    justify-content: center;
    border-top: 1px #777 solid;
}
.ft-social-list li {
    margin: 0.25rem;
    font-size: 1.25rem;
}
.ft-legal {
    background-color: #3f51b5;
}
.ft-legal-list {
    width: 100%;
    display: flex;
    flex-wrap: wrap;
    color: #fff;
}
.ft-legal-list li {
    margin:
        0.125rem 0.625rem;

```



```

    white-space: nowrap;
  }
  .ft-legal-list li:nth-last-child(2) {
    flex: 1;
  }

```

### **Footer.tsx**

```

import React from "react";
import "../Footer.scss"
import FacebookIcon from '@material-ui/icons/Facebook';
import LinkedInIcon from '@material-ui/icons/LinkedIn';
import InstagramIcon from '@material-ui/icons/Instagram';
const Footer = () => {
  return (<div className="footer">
    <footer>
      <section className="ft-main">
        <div className="ft-main-item ft-main-item1">
          <h2 className="ft-title">About</h2>
          <ul>
            <li><a href="#">Details</a></li>
            <li><a href="#">Portfolio</a></li>
            <li><a href="#">Projects</a></li>
          </ul>
        </div>
        <div className="ft-main-item ft-main-item-234">
          <h2 className="ft-title">Resources</h2>
          <ul>
            <li><a href="#">Docs</a></li>
            <li><a href="#">Blog</a></li>
            <li><a href="#">Webinars</a></li>
          </ul>
        </div>
        <div className="ft-main-item ft-main-item-234">
          <h2 className="ft-title">Contact</h2>
          <ul>
            <li><a href="#">Help</a></li>
            <li><a href="https://www.facebook.com/search/top?q=nokia%20timisoara%20careers" target="blank"><FacebookIcon /></a><a href="https://www.linkedin.com/company/nokia/mycompany/verification/" target="blank"><LinkedInIcon /></a><a href="#"><InstagramIcon /></a></li>
          </ul>
        </div>
        <div className="ft-main-item ft-main-item-234">
          <h2 className="ft-title">Stay Updated!</h2>
          <p>Subscribe to our newsletter to get our latest news.</p>
          <form>
            <input type="email" name="email" placeholder="Enter email address" />

```



```

        <input type="submit" value="Subscribe" />
      </form>
    </div>
  </section>
  <section className="ft-legal">
    <ul className="ft-legal-list">
      <li><a href="#">Terms & Conditions</a></li>
      <li><a href="#">Privacy Policy</a></li>
      <li>&copy; 2021 Nokia Networks Romania</li>
    </ul>
  </section>
</footer>
</div>
)
}
export default Footer;

```

### Header.scss

```

header {
  background-color: red;
}
.navbar-text {
  margin-right: 10%
}
.header-title {
  color: antiquewhite;
  display: inline-block;
  font-family: Arial, Helvetica, sans-serif;
  font-style: oblique;
}
.log-button {
  width: 150px;
  height: 40px;
  margin-right: 10px;
}
.avatar {
  background: #3853eb;
  border-radius: 50%;
  height: 50px;
  position: relative;
  width: 50px;
}
.icon {
  height: 50%;
  left: 50%;
  position: absolute;
  stroke-width: 0;
  top: 50%;
  transform: translate(-50%,

```



```

-50%);
  width: 50%;

}
.avatar-name {
  max-width: 100px;
  padding-left: -10px;
  font-size: 20px;
}
.MuiButton-label {
  color: white;
}
@media only screen and (max-width: 1000px) {
  .navbar-brand{
    display: none;
    height: 120px;
  }
  .avatar {
    width: 50px;
  }
  .avatar-name {
    display: none;
    padding-left: 10px;
  }
}
@media only screen and (min-width: 1001px) {
  .navbar-brand{
    margin-left: 100px;
  }
  .avatar {
    height: 45px;
    width: 175px;
    border-radius: 20px 20px;
  }
  .icon {
    margin: auto auto 50px -20px;
    height: 45px; } }

```

### **Header.tsx**

```

import { Link } from "react-router-dom";
import "../Header.scss"
import React, { useEffect, useState } from "react";
import { config } from "../../config"
import { useSelector, useDispatch } from "react-redux";
import { AUTH_SIGN_IN, AUTH_SIGN_OUT, AUTH_ERROR } from '../../redux/reducers/types'
import { ExitToApp, ThreeDRotation } from '@material-ui/icons';
import { withStyles } from '@material-ui/core/styles';
import GeneralModal from "../../designSystems/Modal";
import List from '@material-ui/core/

```





```

List';
import Divider from '@material-ui/core/Divider';
import LinkedInIcon from '@material-ui/icons/LinkedIn';
import Sidebar from '../SideBar/SideBar'
import { withTranslation } from "react-il8next";
import "../services/il8n";
import LanguageSelector from "../LanguageSelector";
import { useTranslation } from "react-il8next";
import { createStyles, makeStyles, Theme } from '@material-ui/
core/styles';
import AppBar from '@material-ui/core/AppBar';
import Toolbar from '@material-ui/core/Toolbar';
import Button from '@material-ui/core/Button';
import Typography from '@material-ui/core/Typography';
import IconButton from '@material-ui/core/IconButton';
import Avatar from '@material-ui/core/Avatar';
import MenuItem from '@material-ui/core/MenuItem';
import Menu from '@material-ui/core/Menu';
import GenericModal from "../designSystems/genericModal";
const useStyles = makeStyles((theme: Theme) =>
  createStyles({
    root: {
      flexGrow: 1,
    },
    customizeToolbar: {
      minHeight: 60
    },
    small: {
      width: theme.spacing(3),
      height: theme.spacing(3),
    },
    menuButton: {
      marginRight: theme.spacing(2),
    },
    title: {
      flexGrow: 1,
    },
  })),
);
export const Header = () => {
  const user = useSelector((state) => ({ auth: state.auth }));
  const [auth, setAuth] = React.useState(true);
  const classes = useStyles();
  const [state, setState] = useState();
  const [modalLoginShow, setModalLoginShow] =
useState<boolean>(false);
  const [pic, setPic] = useState();
  const { t, il8n } = useTranslation();
  const [anchorEl, setAnchorEl] = React.useState<null |
HTMLElement>(null);
  const [anchorElMenu, setAnchorElMenu] = React.useState<null |
HTMLElement>(null);

```



```

    const open = Boolean(anchorEl);
    const handleChange = (event:
React.ChangeEvent<HTMLInputElement>) => {
      setAuth(event.target.checked);
    };
    const handleMenu = (event: React.MouseEvent<HTMLElement>) => {
      setAnchorEl(event.currentTarget);
    };
    const handleClose = () => {
      setAnchorEl(null);
      setAnchorElMenu(null);
    };
    function getModalStyle() {
      return {
        width: '50%',
        maxWidth: '100vw',
        maxHeight: '100%',
        position: 'fixed',
        top: '50%',
        left: '25%',
        transform: 'translate(0, -50%)',
        overflowY: 'auto'
      };
    }

    const handleModal = () => {
      setModalLoginShow(!modalLoginShow)
    }
    const StyledMenu = withStyles({
      paper: {
        border: '1px solid #d3d4d5',
      },
    })((props) => (
      <Menu
        elevation={0}
        getContentAnchorEl={null}
        anchorOrigin={{
          vertical: 'bottom',
          horizontal: 'center',
        }}
        transformOrigin={{
          vertical: 'top',
          horizontal: 'center',
        }}
      />
    ));
    const StyledMenuItem = withStyles((theme) => ({
      root: {
        '&:focus': {
          backgroundColor: theme.palette.primary.main,
          '& .MuiListItemIcon-root, & .MuiListItemText-primary': {
            color:

```



```

theme.palette.common.white,
    },
  },
  },
))) (MenuItem);
const changeLanguage = (event) => {
  i18n.changeLanguage(event.target.value);
};
const dispatch = useDispatch()
useEffect(() => {
  setState({
    authenticated: true,
    user: 'Roxana'
  })
  ;
  sessionStorage.setItem('userEmail', 'roxana.savu')
  sessionStorage.setItem('name', 'Roxana')
  sessionStorage.setItem('token', 'test token')
  sessionStorage.setItem('roles', 'admin')
  dispatch({
    type: AUTH_SIGN_IN,
    payload: {
      role: 'L3',
      authenticated: true,
      name: 'ROXANA',
      email: 'roxana.savu',
      token: 'test token'
    }
  })
  ;
  getIcon('test token')
}, [])
const getIcon = (token) => {
  console.log(token)
  fetch("https://picsum.photos/200/300", {
    method: "GET",
  })
  .then(response => {
    if (response.status === 200) return response;
    throw new Error("failed to authenticate user");
  })
  .then(response => response.blob())
  .then(blob => setPic(URL.createObjectURL(blob)))
  .catch(error => {
    console.log(error)
  });
}
const body = (
  <div>
    <List component="nav" className={classes.root} aria-
label="mailbox-folders">
      <h5>Please login in

```



```

order to continue:</h5>
      <Button variant="contained" color="primary">
        <LinkedInIcon />LinkedIn
      </Button>
      <div className="container">
        <div className="border" />
        <span className="content">
          or
        </span>
        <div className="border" />
      </div>
    </List>
  </div>
);

const _handleSignInClick = () => {
  user.auth.type === "student" ? setModalLoginShow(true) :
  window.open(config.baseURL + config.baseLOCATION + "/auth/
azure", "_self");
}; const _handleLogoutClick = () => {
  window.open(config.baseURL + config.baseLOCATION + "/auth/
logout", "_self");
  sessionStorage.removeItem('exp')
  sessionStorage.removeItem('userEmail')
  sessionStorage.removeItem('name')
  sessionStorage.removeItem('token')
  sessionStorage.removeItem('roles')
  props.handleNotAuthenticated();
};
const _handleNotAuthenticated = () => {
  setState({ authenticated: false });
};
return (
  <div className={classes.root}>
    <AppBar className="nav-bar" position="fixed">
      <Toolbar className={classes.customizeToolbar}>
        { user.auth.type !== "student" ? <Sidebar /> : null }
        <Typography variant="h6" className={classes.title}>
          <Link className="navbar-brand text-white"
to={config.baseLOCATION + "/"}>
            <b>NOKIA</b> {config.AppName} {user.auth.type ===
'student' ? <div className="header-title"> {t("navbar.students")}
</div> : null}
          </Link >
        </Typography>
        {(state && state.authenticated) ? (
          <div className="avatar">
            {pic ?
              <div className='icon'>
                <IconButton
                  aria-label="account of current user"

```



```

        aria-controls="menu-appbar"
        aria-haspopup="true"
        className="icon"
        onClick={handleMenu}
        color="inherit"
      >
      <Avatar alt="avatar Sharp"
        // className={classes.small}
        src={pic}
      />
      <div className='avatar-
name'>{user.auth.name}</div>
    </IconButton>
  </div>
  : null}
  <Menu
    id="menu-appbar"
    anchorEl={anchorEl}
    anchorOrigin={{
      vertical: 'top',
      horizontal: 'right',
    }}
    keepMounted
    transformOrigin={{
      vertical: 'top',
      horizontal: 'right',
    }}
    open={open}
    onClose={handleClose}
  >
    <MenuItem onClick={handleClose}>Profile</MenuItem>
    <MenuItem onClick={_handleLogoutClick}><span
title="log out"><ExitToApp />Log out</span></MenuItem>
  </Menu>

</div>
) : (<div><Button variant="text"
onClick={_handleSignInClick}><span title="log
in">{t("login.label")}</span></Button></div>)}
  </Toolbar>
</AppBar>
{modalLoginShow ?
  <GenericModal
    open={modalLoginShow}
    getModalStyle={getModalStyle}
    title="Sign In Methods"
    handleModal={handleModal}
    handleClose={handleClose}
    body={body}
  /> : null
}
</div>

```



```
);  
}
```

### Homepage.scss

```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  background-color: #2196F3;  
  padding: 10px;  
}  
.grid-item {  
  background-color: rgba(255, 255, 255, 0.8);  
  border: 1px solid rgba(0, 0, 0, 0.8);  
  padding: 20px;  
  font-size: 30px;  
  height: 200px;  
  text-align: center;  
}  
ul {  
  text-align: center;  
}  
.toppane {  
  color: black;  
  margin-top: 80px;  
  margin-left: auto;  
  margin-right: auto;  
  width: 30%;  
  height: 100%;  
}  
.home-page h5 {  
  color: red;  
  font: bold;  
}  
.flexbox-container{  
  position: sticky;  
  margin-top: 2%;  
  display: flex;  
  text-align: center;  
  flex-wrap: wrap-reverse;  
}  
.fill{  
  flex: 1 0 15%;  
}  
.fillx2{  
  flex: 2 0 70%;  
}  
.box-center{  
  width: 300px;  
}  
.box-left{  
  width: 300px;
```



```

}
.box-right{
  width: 300px;
}
.leftpane {
  width: 100%;
  background-color: darkgrey;
  border-collapse: collapse;
}
.middlepane {
  width: 100%;
  background-color: darkgrey;
  height: 100%;
  border-collapse: collapse;
}
.rightpane {
  width: 100%;
  height: 100%;
  background-color: darkgrey;
  border-collapse: collapse;
  text-align: center;
}
.toppane {
  width: 100%;
  height: 40%;
  border-collapse: collapse;
  text-align: center;
}
} .home-container {
  position: relative;
  width: 100%;
  min-height: 550px;
  border-collapse: collapse;
  height: 70%; } .getStarted {
  text-align: center;
  @media screen {.title-text {font-size: 24px;}}
  @media screen and (min-width:1801px) and (max-width:2200px)
{.title-text {font-size: 15px;}}
  @media screen and (min-width:1401px) and (max-width:1800px)
{.title-text {font-size: 14px;}}
  @media screen and (min-width:1125px)and (max-width:1400px)
{.title-text {font-size: 13px;}}
  @media screen and (min-width:901px) and (max-width:1125px)
{.title-text {font-size: 12px;}}
  @media screen and (min-width:801px) and (max-width:900px)
{.title-text {font-size: 11px;}}
  @media screen and (min-width:670px)and (max-width:800px)
{.title-text {font-size: 10px;}}
  @media screen and (min-width:560px)and (max-width:670px)
{.title-text {font-size: 9px;}}
  @media screen and (min-width:512px) and (max-width:560px)
{.title-text {font-size: 8px;}}
  @media screen and (min-

```



```
width:370px) and (max-width:511px){.title-text {font-size: 6px;}}
  @media screen and (max-width:369px){.title-text {font-size:
4px;}}
  .sun-text {
    font-family: Arial, Helvetica, sans-serif;
    text-align: center;
    position: absolute;
    bottom: 50%;
    width: 15vw;
    color: rgb(255, 255, 255);
    text-shadow: 2px 2px #d3c761;
  }
  @media screen and (min-width:1000px){.sun-text {display:
none;}}
  @media screen and (min-width:901px) and (max-width:999px)
{.sun-text {font-size: 18px;}}
  @media screen and (min-width:801px) and (max-width:900px)
{.sun-text {font-size: 16px;}}
  @media screen and (min-width:670px)and (max-width:800px){.sun-
text {font-size: 14px;}}
  @media screen and (min-width:560px)and (max-width:670px){.sun-
text {font-size: 12px;}}
  @media screen and (min-width:512px) and (max-width:560px)
{.sun-text {font-size: 10px;}}
  @media screen and (min-width:370px) and (max-width:511px)
{.sun-text {font-size: 8px;}}
  @media screen and (max-width:369px){.sun-text {font-size:
6px;}}
}
```

## **Homepage.tsx**

```
import React, { useEffect, useState } from "react";
import "../Homepage.scss"
import "../sun.scss"
import "../../services/i18n";
import { motion } from 'framer-motion'
import { useTranslation } from "react-i18next";
import { useSelector, useDispatch } from "react-redux";
import { useHistory } from "react-router-dom";
import { config } from "../../config";
const Homepage = () => {
  const { t, i18n } = useTranslation();
  const [rotation, setRotation] = useState(0)
  const user = useSelector((state) => ({ auth: state.auth }));
  let rocket = null;
  const changeLanguage = (event) => {
    i18n.changeLanguage(event.target.value);
  };
  const history = useHistory();
```





```

function getCenter(element) {
  const { left, top, width, height } =
element.getBoundingClientRect();
  return { x: left + width / 2, y: top + height / 2 }
}
useEffect(() => {
  const arrowCenter = getCenter(rocket);
  addEventListener("mousemove", ({ clientX, clientY }) => {
    const radAngle = Math.atan2(clientY - arrowCenter.y, clientX
- arrowCenter.x);
    const degAngle = radAngle * (180 / Math.PI)
    degAngle >= 360 ? setRotation(90 + rotation - degAngle) :
setRotation(90 + rotation + degAngle)
  })
}, [])
return (
  <div className="home-page">
    <div className="toppane">
      {!user.auth.isAuthenticated ? (<>
        <h1>Welcome, please login in order to continue!</h1>
      </>
      ) : (
        <div>
          <div className="getStarted">
            <motion.h1
              className="sun"
              initial={{ x: 100, y: -100 }}
              animate={{ x: 0, y: 0 }}
              transition={{ ease: 'easeOut', duration: 1.3 }}
              drag={true}
              dragConstraints={{ top: 1, bottom: 1, right: 1,
left: 1 }}
              dragElastic={1}
            >
              <div className="sun-text">
                <b>NOKIA</b> {config.AppName}
              </div>
            </motion.h1>
            <div className="space">
              <div className="rocket" style={{ transform:
`rotate(${rotation}deg)` }} ref={div => (rocket = div)}>
                <div className="fuselage">
                  <div className="nose"></div>
                  <div className="head">
                    <div className="window"></div>
                  </div>
                  <div className="neck"></div>
                  <div className="body"></div>
                  <div className="reactor"></div>
                  <div className="fire">
                    <div className="spark1"></div>

```



```

        <div className="spark2"></div>
        <div className="spark3"></div>
        <div className="spark4"></div>
        <div className="spark5"></div>
        <div className="spark6"></div>
    </div>
</div>
<div className="left-fin">
</div>
<div className="left-fin-end">
    <div className="rocket-text1">
        {user.auth.name}
    </div>
</div>
<div className="right-fin"></div>
<div className="right-fin-end">
    <div className="rocket-text2">
        "Your Plan, Your Planet"
    </div>
</div>
</div>
</div>
<motion.div animate={{ opacity: [0, 1] }} >
    <div className="circle c1" onClick={e =>
history.push('/projects/list')}><p className="circle-text">Apply
to a project</p></div>
        <div className="circle c1 ripple" />
        <div className="circle c2" onClick={e =>
history.push('/newsletter')}><p className="circle-text">View
newsletter</p></div>
        <div className="circle c2 ripple" />
        <div className="circle c3" onClick={e =>
history.push('/credits')}><p className="circle-text">Credits</p></
div>
        <div className="circle c3 ripple" />
        <div className="circle c4" onClick={e =>
history.push('/projects/list')}><p className="circle-text">Check
application results</p></div>
        <div className="circle c4 ripple" />
        <div className="star star1"></div>
        <div className="star star2"></div>
        <div className="star star3"></div>
        <div className="star star4"></div>
        <div className="star star5"></div>
    </motion.div>
</div>
</div>
)}
{<motion.h5
    initial={{ x: 100, y: -100 }}
    animate={{ x: 0, y: 100 }}
    transition={{ ease:

```



```

'easeOut', duration: 1.3 }}
    >
    <div className="development">This web application is
still under development!</div>
  </motion.h5>}
</div>
</div>)
}
export default Homepage

```

### sun.scss

```

.home-page {
  width: 100%;
  height: 100%;
  display: flex;
  margin: 0;
  padding: 1rem;
  min-height: 100vh;
  background: #a2d4ff;
  background-attachment: fixed;
  background-image:
    radial-gradient(white, rgba(40, 25, 110, 0.2) 2px,
transparent 40px),
    radial-gradient(white, rgba(255,255,255,.15) 1px,
transparent 30px),
    radial-gradient(white, rgba(255,255,255,.1) 2px,
transparent 40px),
    radial-gradient(rgba(255,255,255,.4), rgba(255,255,255,.1)
2px, transparent 30px),
    linear-gradient(#333 ,#2196F3 );
  background-size: 550px 550px, 350px 350px, 250px 250px,
150px 150px,cover;
  background-position: center;
}

.sun {
  top: -22vw;
  left: -22vw;
  min-width: 60%;
  max-width: 90%;
  margin: 3rem auto;
  position: absolute;
  border-radius: 50%;
  -webkit-mask-image: radial-gradient(rgba(0, 0, 0, 1.0) 40%,
transparent 65%);
  mask-image: radial-gradient(rgba(0, 0, 0, 1.0) 40%,
transparent 65%);
  background: radial-gradient(#FFFFFFE0, #FFE4B5 27%, transparent
calc(27% + 3px) 100%),

```



```

    radial-gradient(transparent 50%, white), radial-
gradient(#fff8bd, transparent 70%);
    box-shadow: inset -8px -7px 4px 0px rgba(0,0,0,0.3);
    display: flex;
    justify-content: center;
    &::after{
        content: '';
        display: block;
        padding-bottom: 100%;
    }

    &::before{
        content: '';
        border-radius: 50%;
        width: 50%;
        padding-bottom: 50%;
        padding-bottom: 0vw;
        width: 15vw;
        height: 15vw;
    }
}

.circle {
    min-width: 12vw;
    max-width: 12vw;
    margin: 3rem auto;
    align-items: center;
    justify-content: center;
    box-shadow: inset -8px -7px 4px 0px rgba(0, 0, 0, 0.5),
    rgba(255, 255, 255, 0.486) 0px 54px 55px,
    rgba(255, 255, 255, 0.466) 0px -12px 30px,
    rgba(255, 255, 255, 0.527) 0px 4px 6px,
    rgba(255, 255, 255, 0.562) 0px 12px 13px,
    rgba(255, 255, 255, 0.432) 0px -3px 5px;
    position: absolute;
    border-radius: 50%;
    display: flex;
    transition: transform .7s ease-in-out;
}

.circle:before {
    border-radius: 50%;
    width: 12vw;
    padding-bottom: 12vw;
    content: '';
    padding-bottom: 0vw;
    width: 12vw;
    height: 12vw;
}

.circle-text {
    font-family: Arial, Helvetica, sans-serif;
    font-size: 11px;
    font-weight: bold;

```



```

    text-align: center;
    position: absolute;
    width: 100%;
}
@media all { .circle-text {font-size: 24px;}}
@media all and (min-width:1801px) and (max-width:2200px)
{.circle-text {font-size: 22px;}}
    @media all and (min-width:1401px) and (max-width:1800px)
{.circle-text {font-size: 20px;}}
    @media all and (min-width:1125px)and (max-width:1400px){.circle-
text {font-size: 18px;}}
    @media all and (min-width:901px) and (max-width:1125px){.circle-
text {font-size: 16px;}}
    @media all and (min-width:801px) and (max-width:900px){.circle-
text {font-size: 14px;}}
    @media all and (min-width:670px)and (max-width:800px){.circle-
text {font-size: 12px;}}
    @media all and (min-width:560px)and (max-width:670px){.circle-
text {font-size: 10px;}}
    @media all and (min-width:480px) and (max-width:560px){.circle-
text {font-size: 8px;}}
    @media all and (min-width:370px) and (max-width:479px){.circle-
text {font-size: 6px;}}
    @media all and (max-width:369px){.circle-text {font-size:
4px;}}
    .c1 {
        top: 25vw;
        left: 10vw;
        background: #D2B48C;
        border-color: #DEB887;
        background: linear-gradient(
            #D2B48C,
            #F5DEB3,
            #D2B48C,
            #F5DEB3,
            #D2B48C,
            #F5DEB3
        );
        cursor: pointer;
        z-index: 10;
    }

    .c2 {
        cursor: pointer;
        z-index: 10;
        top: 3vw;
        left: 28vw;
        background: #7449ec85;
        border-color: #ec594985;
        background: linear-gradient(
            #f8ada5,
            #fca298,

```



```

        #fc968b,
        #f8ada5,
        #fca298,
        #fc968b,
        #f8ada5,
        #fca298,
        #fc968b
    );
}
.c3 {
    cursor: pointer;
    z-index: 10;
    top: 6vw;
    left: 70vw;
    background: rgb(172, 154, 98), ;
    border-color: #e7d08f;
    background: linear-gradient(
        #FFE4C4,
        #FFEBCD,
        #FFF8DC,
        #FFE4C4,
        #FFEBCD,
        #FFF8DC,
        #FFE4C4,
        #FFEBCD,
        #FFF8DC,
        #FFE4C4,
    );
}
.c4 {
    cursor: pointer;
    z-index: 10;
    top: 27vw;
    left: 80vw;
    background: #78dde0;
    border-color: #75d1d1;
    background: radial-gradient(#a3dee0, #4e9396 );
}
.ripple {
    transform: scale(2);
    background: none;
    animation-name: rippleAnimation;
    animation-duration: 1.5s;
    animation-iteration-count: infinite;
    cursor: auto ;
    z-index: 1;
}
@keyframes rippleAnimation {
    0% {
        transform: scale(1.5);
    }
}

```



```

    70% {
      transform: scale(2.5);
      opacity: 0;
    }
    100% {
      transform: scale(0);
      opacity: 0;
    }
  }
  .development{
    margin: 3rem auto;
    align-items: center;
    justify-content: center;
    position: absolute;
    display: flex;
    top: 23vw;
    left:14vw;
    width: 70vw;
    height: aoto;
  }
  .star {
    position: absolute;
    display: flex;
    margin: 3rem auto;
    height: 0.3vw;
    background: linear-gradient(-45deg, #082d7c,
    rgba(255, 0, 0, 0));
    border-radius: 2vw;
    filter: drop-shadow(0 0 1vw #082d7c);
    animation: tail 3s ease-in-out infinite,
      falling 3s ease-in-out infinite;
  }

  @keyframes tail {
    0% {
      width: 0;
    }
    30% {
      width: 7vw;
    }
    100% {
      width: 0;
    }
  }
  @keyframes falling {
    0% {
      transform: translateX(0);
    }
    100% {
      transform: translateX(15vw);
    }
  }

```



```

}
.star::before,
.star::after {
  content: '';
  position: absolute;
  top: -0.8vw;
  right: 0;
  height: 2vw;
  background: linear-gradient(-45deg, rgba(0, 0,
255, 0), #cfdaf1, rgba(0, 0, 255, 0));
  border-radius: 5vw;
  transform: translateX(50%) rotateZ(45deg);
  animation: shining 3s ease-in-out infinite;
}
@keyframes shining {
  0% {
    width: 0;
  }
  50% {
    width: 0.3vw;
  }
  100% {
    width: 0;
  }
}
.star::after {
  transform: translateX(50%) rotateZ(-45deg);
}
.star1 {
  top: 20vw;
  left: 20vw;
  animation-delay: 1s;
}
.star1::before,
.star1::after {
  animation-delay: 1s;
}
.star2 {
  top: 25vw;
  left: 75vw;
  animation-delay: 1.2s;
}
.star2::before,
.star2::after {
  animation-delay: 1.2s;
}
.star3 {
  top: 5vw;
  left: 80vw;
  animation-delay: 1.4s;
}

```





```

        .star3::before,
        .star3::after {
            animation-delay: 1.4s;
        }
        .star4 {
            top: 40vw;
            left: 75vw;
            animation-delay: 1.6s;
        }
        .star4::before,
        .star4::after {
            animation-delay: 1.6s;
        }
        .star5 {
            top: 40vw;
            left: 23vw;
            animation-delay: 1.8s;
        }
        .star5::before,
        .star5::after {
            animation-delay: 1.8s;
        }
    .rocket-text1 {
        font-family: Arial, Helvetica, sans-serif;
        font-weight: bold;
        text-align: center;
        position: absolute;
        font-size: 11px;
        width: 100%;
        top: 7vw;
        left: -5vw;
        color: #000;
        -webkit-transform: rotate(-45deg);
        -moz-transform: rotate(-45deg);
    }
    .rocket-text2 {
        font-family: Arial, Helvetica, sans-serif;
        font-weight: bold;
        position: absolute;
        font-size: 8px;
        width: 100%;
        top: 4vw;
        left: 3.5vw;
        color: #000;
        display: inline-block;
        white-space: nowrap;
        -webkit-transform: rotate(44deg);
        -moz-transform: rotate(44deg);
    }
    .space {
        top: 4vw;
        left: 35vw;
    }

```



```

    position: relative;
    display: block;
    height: 30vw;
    width: 30vw;
    color: white;
    //padding-top: 5vw;
    float: left;
    margin: 1.5%;
    // cursor: pointer;
    background: transparent;
    align-items: center;
    align-content: center;
}
.rocket {
    cursor: pointer;
    position: relative;
    width: 10vw;
    height: 20vw;
    margin: 0 auto;
    transition: transform .2s ease-in-out;
    background-color: transparent;
    align-items: center;
    align-content: center;
    z-index: 100;
}
.fuselage {
    margin: 0 auto;
    height: auto;
}
.nose {

    width: 2vw;
    margin: 0 auto;
    border-top: 1vw solid #353535;
    border-top-left-radius: 30vw;
    border-top-right-radius: 30vw;
    height: auto;
}
.head {
    display: block;
    border-bottom: 3vw solid white;
    border-left: 1.3vw solid transparent;
    border-right: 1.3vw solid transparent;
    height: 0;
    width: 5vw;
    margin: 0 auto;
}
.window {
    margin: 0 auto;
    position: relative;
    top: 0.5vw;

```



```

    left:-0.5vw;
    border-bottom: 1.5vw solid #00a0d6;
    border-left: 0.5vw solid transparent;
    border-right: 0.5vw solid transparent;
    height: 0vw;
    width: 3.5vw;
  }
  .neck{
    position: relative;
    width: 5vw;
    height: 2.5vw;
    margin: 0 auto;
    background: #f4f4f4;
    &:after{
      margin: 0 auto;
      position: absolute;
      top: 0vw;
      right: 0vw;
      content: '';
      width: 2.5vw;
      height: 4vw;
      background: #e0e0e0;
    }
  }
  .body{
    margin: 0 auto;
    position: relative;
    width: 5vw;
    height: 13vw;
    background: #f4f4f4;
    border-bottom: 0.5vw solid #a81237;
    border-top: 0.5vw solid #a81237;
    &:after{
      margin: 0 auto;
      position: absolute;
      //top: 0vw;
      //right: 0vw;
      content: '';
      width: 2.5vw;
      height: 12vw;
      background: #e0e0e0;
    }
  }
  .reactor {
    position: relative;
    border-bottom: 2.5vw solid #4f4f4f;
    border-left: 1vw solid transparent;
    border-right: 1vw solid transparent;
    height: 0vw;
    width: 5.8vw;
    margin: 0 auto;
    &:after {

```



```

        margin: 0 auto;
        content: '';
        position: absolute;
        // top: -2.5vw;
        left: -1vw;
        width: 3vw;
        opacity: 0.3;
        border-bottom: 2.5vw solid #363636;
        border-left: 1vw solid transparent;
    }
}
.fire {
    position: relative;
    width: 3vw;
    height: 5vw;
    margin: 0 auto;
    border-bottom-right-radius: 5vw;
    border-bottom-left-radius: 5vw;
    animation: fire 0.2s infinite;
    > div {
        position: absolute;
        background: rgb(34, 141, 212);
    }
}
.spark1 {
    top: 2vw;
    left: 1.5vw;
    width: 23vw;
    height: 23vw;
    border-radius: 50%;
    animation: fireBig 0.5s infinite;
}
.spark2 {
    top: 3.5vw;
    right: 0.5vw;
    width: 25vw;
    height: 25vw;
    border-radius: 50%;
    animation: fireBig 0.3s infinite;
}
.spark3 {
    top: 3vw;
    right: 1.5vw;
    width: 25vw;
    height: 25vw;
    border-radius: 50%;
    animation: fireSmall 0.4s infinite;
}
.spark4 {
    top: 2vw;
    left: 1vw;
    width: 23vw;

```



```

        height: 23vw;
        border-radius:50%;
        animation: fireSmall 0.7s infinite;
    }
    .spark5 {
        top: 1.5vw;
        right: 2.5vw;
        width: 22vw;
        height: 22vw;
        border-radius:50%;
        animation: fireSmall 0.6s infinite;
    }
    .spark6 {
        top: 3.5vw;
        left: 1.5vw;
        width: 28vw;
        height: 28vw;
        border-radius:50%;
        animation: fireSmall 0.2s infinite;
    }
    .left-fin {
        margin: 0 auto;
        position: relative;
        top : -20vw;
        left: -5vw;
        width: 5vw;
        border-bottom: 11.5vw solid white;
        border-left: 4vw solid transparent;
        &-end{
            margin: 0 auto;
            position: relative;
            top : -30vw;
            left: -7.5vw;
            width: 5vw;
            border-bottom: 10vw solid white;
            border-left: 10vw solid transparent;
            border-top-left-radius: 60%;
            border-bottom-left-radius: 10%;
        }
    }
    .right-fin {
        margin: 0 auto;
        position: relative;
        top : -41.5vw;
        left: 5vw;
        width: 5vw;
        border-bottom: 11.5vw solid white;
        border-right: 4vw solid transparent;

        &-end{

```



```

margin: 0 auto;

    position: relative;
    top : -51.5vw;
    left: 7.8vw;
    width: 5vw;
    border-bottom: 10vw solid white;
    border-right: 10vw solid transparent;
    border-top-left-radius: 60%;
    border-bottom-right-radius: 10%;
}

    &:after {
    margin: 0 auto;
    content: '';
    z-index: 2;
    position: absolute;
    top : -0.01vw;
    left: 0vw;
    width: 1.5vw;
    border-bottom: 11.57vw solid #c4c4c4;
    border-right: 1.2vw solid transparent;
    }
}

@keyframes fire {
    0% {
        background: linear-gradient(to bottom, rgb(92,
12, 219) 0%, rgb(7, 61, 160) 50%, rgb(34, 141, 212) 100%);
    }
    50% {
        background: linear-gradient(to bottom, rgb(34,
141, 212) 0%, rgb(92, 12, 219) 51%, rgb(7, 61, 160) 100%);
    }
    100% {
        background: linear-gradient(to bottom, rgb(7,
61, 160) 0%, rgb(34, 141, 212) 51%, rgb(92, 12, 219) 100%);
    }
}

@keyframes fireBig {
    0% {
        width: 2vw;
        height: 2vw;
        background: rgb(92, 12, 219);
    }
    50% {
        background: rgb(34, 141, 212);
    }
    100% {
        width: 6vw;
        height: 6vw;
    }
}
}

```



```

@keyframes fireSmall {
  0% {
    width: 2vw;
    height: 2vw;
    background: rgb(92, 12, 219);
  }
  50% {
    background: rgb(7, 61, 160);
  }
  100% {
    width: 4vw;
    height: 4vw;
  }
}

@keyframes stars {
  0% {
    top: -100%;
  }
  100% {
    top: 0;
  }
}

```

### Landingpage.jsx

```

import React, { useState } from "react";
import { useHistory } from "react-router-dom";
import { UPDATE_PROFILE_TYPE } from '../redux/reducers/types'
import { useSelector, useDispatch } from "react-redux";
import { config } from "../config"
import { motion } from 'framer-motion'
import { useIsSmall } from '../hooks/utils'
import Logout from "../middlewares/Logout"
import "../Landingpage.scss"
import Button from '@material-ui/core/Button';
import {
  Card, CardImg, CardText, CardBody,
  CardTitle, CardSubtitle
} from 'reactstrap';
const Landingpage = () => {
  const dispatch = useDispatch();
  const history = useHistory();
  const user = useSelector((state) => ({ auth: state.auth }));
  const isSmall = useIsSmall()
  const variants = isSmall
    ? {
      animate: {

```



```

        opacity: 1,
        scale: 1,
        y: 0,
      },
      exit: {
        opacity: 1,
        scale: 1,
        y: 500,
      },
    },
  ],
  : {
    animate: {
      opacity: 1,
      scale: 1,
      y: 0,
    },
    exit: {
      opacity: 0,
      scale: 0.9,
      y: -10,
    },
  },
};
const cardClick = (resource_origin) => {
  dispatch({
    type: UPDATE_PROFILE_TYPE,
    payload: {
      type: resource_origin,
    },
  });
  sessionStorage.setItem('type', resource_origin);
  history.push('/home');
}
const handleLogout = () => {
  Logout()
}
return (
  <div className="home-container">

    <div>
      {!user.auth.token ? (
        <div>

          <div className="card-container">
            <motion.a style={{ cursor: 'pointer' }} onClick={()
=> { cardClick('student') }}
              initial={{opacity: 0, y: 180}}
              animate={isSmall ? {opacity: [0, 1], y: [140, 0],
x: [0, 0]} : {opacity: [0, 1], x: [-500, 0]}}
              transition={{ ease: "easeOut", delay: 0.7,
duration: 0.7 }}
            >
              <Card

```





```

className="card-item">
    <CardBody>
        <CardTitle className="card-title">student</
CardTitle>
        <CardText className="text-muted">You are
student at University</CardText>
    </CardBody>
</Card>
</motion.a>
<motion.a style={{ cursor: 'pointer' }} onClick={()
=> { cardClick('nokia') }}
    initial={{opacity: 0, y: 180}}
    animate={isSmall ? {opacity: [0, 1], y: [140, 0],
x: [0, 0]} : {opacity: [0, 1], x: [500, 0]}}
    transition={{ ease: "easeOut", delay: 0.7,
duration: 0.7 }}
    >
        <Card className="card-item">
            <CardBody>
                <CardTitle className="card-title">Nokia
tentant</CardTitle>
                <CardText className="text-muted">You are a
nokia employee / guest </CardText>
            </CardBody>
        </Card>
    </motion.a>
</div>
<motion.p
    animate={isSmall ? { y: 150} : { y: 150}}
    transition={{ ease: 'easeIn', duration: 0.5 }}
    >
        Please select one of the cards in order to continue:
    </motion.p>
</div>
) : (
    <div>
        <h1>{user.auth.name}, you have been logeed in
succcessfully!</h1>
        <Button variant="contained" color="primary"
className="button" onClick={() => { cardClick(user.auth.type) }}
>Continue here</Button>
        <Button variant="contained"
color="secondary" className="button" onClick={() =>
{ handleLogout() }}>Logout</Button>
    </div>
    </div>
    </div>
    </div>
);
}
export default Landingpage;

```



## Landingpage.scss

```
.home-container {
  position: relative;
  width: 60%;
  height: 60%;
  z-index: 15;
  text-align: center;
}
.card-item {
  width: 300px;
  height: 150px;
  flex-shrink: 20;
}
.card-item :hover {
  zoom: 105%;
  transition: 10ms;
}
.card-container{
  position: absolute;
  top:auto;
  bottom: auto;
  display: flex;
  justify-content: space-evenly;
  align-items: center;
  height: 100%;
  width: 100%;
  flex-flow: row wrap;
  flex-shrink: 20;
  flex-wrap: wrap-reverse;
}
.card-title {
  font-weight: bold;
  text-transform: uppercase;
  margin-top: 30px;
  ;
}
.home-container button {
  width: 200px;
  margin: 50px 50px 50px 50px;}
```

## newsletter.scss

```
.home-page-newsletter {
  width: 100%;
  height: 100%;
  display: flex;
  margin: 0;
  padding: 1rem;
  min-height: 100vh;
  background: #a2d4ff;
```



```

background-attachment: fixed;
background-image:
    radial-gradient(rgb(19, 170, 216), rgba(57, 86, 250, 0.767)
2px, transparent 40px),
    radial-gradient(rgb(19, 170, 216), rgba(71, 124, 238, 0.904)
1px, transparent 30px),
    radial-gradient(rgb(19, 170, 216), rgba(77, 195, 250, 0.849)
2px, transparent 40px),
    radial-gradient(rgba(19, 170, 216,.4), rgba(25, 102, 202,
0.726) 2px, transparent 30px),
    radial-gradient(rgba(19, 170, 216,.4), rgba(36, 141, 211,
0.726) 2px, transparent 30px),
    radial-gradient(rgb(59, 161, 230), rgb(34, 141,
212) ,#3049d6, #0820ac,#06156e ,#040d42,#02092e);
background-size: 350px 350px, 300px 300px, 250px 250px,
200px 200px, 160px 160px,cover;
background-position: center;
}
html, body{
    display: border-box;
}
.container {
    z-index: 1;
    perspective:3000px;
}
.container .book {
    cursor: pointer;
    left: 6vw;
    top: -5vw;
    position: relative;
    display: flex;
    width: 20vw;
    height: 25vw;
    margin: 3 auto;
    border-radius: 2px 4px 4px 2px;
    background: linear-gradient(45deg, rgb(10, 56, 155) 0%,
rgb(10, 56, 155) 100%);//#389bf8
    font-family: acumin-pro, sans-serif;
    -webkit-box-shadow: 13px 13px 8px 0px rgba(1, 21, 70, 0.822);
    -moz-box-shadow: 13px 13px 8px 0px rgba(1, 21, 70, 0.822);
    box-shadow: 13px 13px 8px 0px rgba(1, 21, 70, 0.822);
    font-weight: 40;
    color: #ffffff;
    -webkit-transform-style: preserve-3d;
    -moz-transform-style: preserve-3d;
    transform-style: preserve-3d;
    -webkit-transition: -webkit-transform .5s;
    -moz-transition: -moz-transform .5s;
    transition: transform .5s;
}
.container .book:hover {
    -webkit-
        transform:

```



```

rotate3d(0,1,0,35deg);
  -moz-transform: rotate3d(0,1,0,35deg);
  transform: rotate3d(0,1,0,35deg);
}
.container .book > div,
.container .front > div {
  display: block;
  position: absolute;
}
.container .front {
  -webkit-transform-style: preserve-3d;
  -moz-transform-style: preserve-3d;
  transform-style: preserve-3d;
  -webkit-transform-origin: 0% 50%;
  -moz-transform-origin: 0% 50%;
  transform-origin: 0% 50%;
  -webkit-transition: -webkit-transform .5s;
  -moz-transition: -moz-transform .5s;
  transition: transform .5s;
  -webkit-transform: translate3d(0,0,20px);
  -moz-transform: translate3d(0,0,20px);
  transform: translate3d(0,0,20px);
  z-index: 10;
}
.container .front > div {
  width: 20vw;
  height: 25vw;
}
.container .left-side{
  width: 3vw;
  left: -1.5vw;
  height: 25vw;
  background-color: rgb(10, 56, 155);
  -webkit-transform: rotate3d(0,1,0,-90deg);
  -moz-transform: rotate3d(0,1,0,-90deg);
  transform: rotate3d(0,1,0,-90deg);
}
.cover .num-up{
  position: absolute;
  top: 20vw;
  left: 9vw;
  font-size: 1.3vw;
  font-family: "Lucida Console", Courier , monospace;
}
.cover .num-down{
  position: relative;
  top: 6.5vw;
  left: -9.5vw;
  transform: rotate(-90deg);
}
.author{
  font-family: "Lucida Console",

```



```

Courier , monospace;
  font-weight: 4;
  position: absolute;
  top: 5vw;
  left: 5vw;
  opacity: .8;
}
.container .front > div {
  border-radius: 0 3px 3px 0;
  box-shadow:
    inset 4px 0 10px rgba(0, 0, 0, 0.1);
}
.container .front:after {
  content: '';
  position: absolute;
  top: 1vw;
  bottom: 1vw;
  left: -1vw;
  width: 1vw;
}
.container .cover:after {
  content: '';
  position: absolute;
  top: 0;
  left: 1vw;
  bottom: 0;
  width: 0.3vw;
  background: rgba(2, 33, 119, 0.5);
  box-shadow: 1px 0 3px rgba(54, 86, 173, 0.678);
}
.container h2 {
  font-family: "Lucida Console", Courier , monospace;
  width: 20vw;
  height: 4vw;
  color: #ffffff;
  font-size: 1.5vw;
  line-height: 4vw;
  padding-right: 1vw;
  text-align: right;
  -webkit-transform-origin: 0 0;
  -moz-transform-origin: 0 0;
  transform-origin: 0 0;
  -webkit-transform: rotate(90deg) translateY(-40px);
  -moz-transform: rotate(90deg) translateY(-40px);
  transform: rotate(90deg) translateY(-40px);
}
.cover {
  background: radial-gradient(rgba(36, 154, 250) 5%, rgba(13, 139, 243) 15%, rgba(0, 20, 85) );
}
.left-side h2 span:first-child {
  font-weight: 4;
}

```



```

    font-size: 1.3vw;
    padding-right: 2vw;
  }
  .left-side h2 span:last-child{
    font-family: acumin-pro, sans-serif;
  }

  @media only screen and (min-width: 421px) and (max-width: 600px)
  {
    .home-page-newsletter {
      display: block;
    }
    .container .book {
      left: 35vw;
      top: 10vw;
    }
    .container{
      padding-top: 5vw;
    }
  }
  @media only screen and (max-width: 420px){
    .home-page-newsletter {
      display: block;
    }
    .container .book {
      left: 35vw;
      top: 10vw;
    }
    .container{
      padding-top: 10vw;
    }
  }
}

```

### **Newsletter.tsx**

```

import React from "react";
import "./newsletter.scss"
import { useHistory } from "react-router-dom";
import data from "../jNewsletter.json";
export const Newsletter = () => {
  const history = useHistory();

  return (
    <div className="home-page-newsletter">
      {data.map((postData) => {
        console.log(postData);
        return (
          <div className="container " >
            <div className="book" onClick={e => history.push('/firstarticol')}>

```



```

        <div className="front">
          <div className="cover">
            <p className="num-up">{postData.auror1}</p>

            <p className="author">{postData.titlu1}</p>
          </div>
        </div>
        <div className="left-side">
          <h2>
            <span>{postData.data1}</span>
          </h2>
        </div>
      </div>
    </div>
  </div>
  );
}
}
{data.map((postData) => {
  console.log(postData);
  return (
    <div className="container" >
      <div className="book" onClick={e => history.push('/
secondarticol2')}>
        <div className="front">
          <div className="cover">
            <p className="num-up">{postData.auror2}</p>

            <p className="author">{postData.titlu2}</p>
          </div>
        </div>
        <div className="left-side">
          <h2>
            <span>{postData.data2}</span>
          </h2>
        </div>
      </div>
    </div>
  );
}
}
{data.map((postData) => {
  console.log(postData);
  return (
    <div className="container" >
      <div className="book" onClick={e => history.push('/
thirdarticol')}>
        <div className="front">
          <div className="cover">
            <p className="num-up">{postData.auror3}</p>

            <p className="author">{postData.titlu3}</p>
          </div>
        </div>
        <div
          className="left-

```



```

side">
        <h2>
            <span>{postData.data3}</span>
        </h2>
    </div>
</div>
</div>
);
}}
    </div>

)
}

```

### jNewsletter.json

```

[
  {
    "titlu1": "Titlu1 titlu1 titlu1 titlu1",
    "auror1": "Nume autor1",
    "data1": "01.01.2021",
    "titlu2": "Titlu2 titlu2 titlu2 titlu2",
    "auror2": "Nume autor2",
    "data2": "02.02.2021",
    "titlu3": "Titlu3 titlu3 titlu3 titlu3",
    "auror3": "Nume autor3",
    "data3": "03.03.2021"
  }
]

```

### FirstArticol.scss

```

.corp {
  width: 100%;
  height: 100%;
  display: flex;
  margin: 0;
  padding: 1rem;
  min-height: 100vh;
  background: #a2d4ff;
  background-attachment: fixed;
  background-image:
    radial-gradient(rgba(19, 170, 216), rgba(57, 86, 250, 0.767)
2px, transparent 40px),
    radial-gradient(rgba(19, 170, 216), rgba(71, 124, 238, 0.904)
1px, transparent 30px),
    radial-gradient(rgba(19, 170, 216), rgba(77, 195, 250, 0.849)
2px, transparent 40px),

```





```

        radial-gradient(rgba(19, 170, 216,.4), rgba(25, 102, 202,
0.726) 2px, transparent 30px),
        radial-gradient(rgba(19, 170, 216,.4), rgba(36, 141, 211,
0.726) 2px, transparent 30px),
        radial-gradient(rgb(59, 161, 230), rgb(34, 141,
212) ,#3049d6, #0820ac,#06156e ,#040d42,#02092e);
        background-size: 350px 350px, 300px 300px, 250px 250px,
200px 200px, 160px 160px,cover;
        background-position: center;
        --page-bg: #f5f5f5;
        --dark-text: #2A2935;
        --baseline: 1.2vw;
        --base-size: var(--baseline) * 1.2;
    }
    * {
        box-sizing: border-box;
    }
    .corp {
        justify-content: center;
    }
    .articol-cover {
        position: relative;
        top: 8vw;
        width: calc(var(--baseline) * 60);
        height: calc(var(--baseline) * 25);
        box-shadow: 0 0 100px rgb(0, 0, 0);
    }
    .book {
        width: 100%;
        height: 100%;
        display: flex;
        perspective: 1200px;
        &__page {
            position: relative;
            width: 50%;
            height: 100%;
            display: grid;
            transform: rotateY(0deg);
            transition: transform 0.9s cubic-bezier(0.645, 0.045, 0.355,
1);
            transform-origin: 0% 0%;
            background-color: var(--page-bg);
            background-image: linear-gradient(90deg, rgb(199, 197, 197)
0%, rgba(247,247,247,0) 18%);
            &:nth-of-type(1) {
                background-image: linear-gradient(-90deg, rgb(199, 197, 197)
0%, rgba(10, 10, 10, 0) 18%);
            }
            &--1 {
                cursor: pointer;
                overflow: hidden;
            }
        }
    }

```



```

    &--2 {
      position: absolute;
      right: 0;
      pointer-events: none;
      transform-style: preserve-3d;
      background-color: var(--page-bg);
      background-image: linear-gradient(90deg, rgb(199, 197, 197)
0%, rgba(247,247,247,0) 18%);
    }
    &--4 {
      cursor: pointer;
      padding: 0 calc(var(--baseline) * 3);
    }
    &-front {
      position: absolute;
      width: 100%;
      height: 100%;
      transform: rotateY(0deg) translateZ(1px);
    }
    &-back {
      position: absolute;
      width: 100%;
      height: 100%;
      padding: 0 calc(var(--baseline) * 1.8);
      transform: rotateY(180deg) translateZ(1px);
    }
  }
  input[type="radio"] {
    display: none;
    &:checked+.book__page {
      transition: transform 0.9s cubic-bezier(0.645, 0.045, 0.355,
1);
      transform: rotateY(-180deg);
    }
  }
}
.titluart1{
  position: relative;
  font-family:"Lucida Console", Courier , monospace ;
  font-size: 3vw;
  top: 10vw;
  left: 5vw;
}
.text{
  position: relative;
  font-family:"Lucida Console", Courier , monospace ;
  font-size: 1.5vw;
  top: 1.5vw;
  left: 2vw;
}
.autorart1{
  position:
relative;

```



```

    font-family:"Lucida Console", Courier , monospace ;
    top: 5vw;
    left: 15vw;
}
.dataart1{
    position: relative;
    font-family:"Lucida Console", Courier , monospace ;
    top: 5vw;
    left: 2.5vw;
}
@media only screen and (max-width: 1200px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 30);
        top: 10vw;
    }
}
@media only screen and (max-width: 1100px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 35);
        top: 10vw;
    }
}
@media only screen and (max-width: 1000px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 40);
        top: 10vw;
    }
}
@media only screen and (max-width: 900px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 45);
        top: 10vw;
    }
}
@media only screen and (max-width: 855px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 37);
        top: 15vw;
    }
}
@media only screen and (max-width: 800px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 40);
        top: 15vw;
    }
}
}

```



```

@media only screen and (max-width: 700px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 45);
    top: 15vw;
  }
}
@media only screen and (max-width: 640px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 50);
    top: 15vw;
  }
}
@media only screen and (max-width: 600px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 70);
    top: 20vw;
  }
}
@media only screen and (max-width: 550px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 75);
    top: 20vw;
  }
}
@media only screen and (max-width: 520px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 80);
    top: 20vw;
  }
}
@media only screen and (max-width: 470px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 90);
    top: 20vw;
  }
}

```

### **FirstArticol.tsx**

```

import React from "react";
import {useState,useEffect} from 'react';
import "./FirstArticol.scss"
import data from "../jFirstArticoll1.json"
export const FirstArticol = () =>{
  return(
    <div className="corp">
      {data.map((postData) => {

```



```

console.log(postData);
return (
  <div className="articol-cover" >
<div className="book">
<label htmlFor="page-1" className="book__page book__page--1">
  <div className="titluart1">{postData.titluart1}</div>
  <div className="autorart1">{postData.autorart1}</div>
  <div className="dataart1">{postData.dataart1}</div>
</label>
<label htmlFor="page-2" className="book__page book__page--4">
  <div className="content">
    <div className="text">
      {postData.pagina3}
    </div>
  </div>
</label>
<input type="radio" name="page" id="page-1"/>
<input type="radio" name="page" id="page-2"/>
<label className="book__page book__page--2">
  <div className="book__page-front">
    <div className="content">
      <div className="text">{postData.pagina1}</div>
    </div>
  </div>
  <div className="book__page-back">
    <div className="content">
      <div className="text">{postData.pagina2}</div>
    </div>
  </div>
</label>
</div>
</div>
);
}})
  </div>
)
}

```

### jFirstArticol1.json

```

[
  { "titluart1": "Titlul titlul titlul titlul",
    "autorart1": "Autor1",
    "dataart1": "01.01.2021",
    "pagina1": "Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1
Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1
Pagina1 Pagina1 Pagina1 Pagina1",
    "pagina2": "Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2
Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2
Pagina2 Pagina2 Pagina2 Pagina2",

```



```

        "pagina3": "Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3
Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3
Pagina3 Pagina3 Pagina3 Pagina3 "
    }
]

```

## SecondArticol.scss

```

.corp {
  width: 100%;
  height: 100%;
  display: flex;
  margin: 0;
  padding: 1rem;
  min-height: 100vh;
  background: #a2d4ff;
  background-attachment: fixed;
  background-image:
    radial-gradient(rgb(19, 170, 216), rgba(57, 86, 250, 0.767)
2px, transparent 40px),
    radial-gradient(rgb(19, 170, 216), rgba(71, 124, 238, 0.904)
1px, transparent 30px),
    radial-gradient(rgb(19, 170, 216), rgba(77, 195, 250, 0.849)
2px, transparent 40px),
    radial-gradient(rgba(19, 170, 216,.4), rgba(25, 102, 202,
0.726) 2px, transparent 30px),
    radial-gradient(rgba(19, 170, 216,.4), rgba(36, 141, 211,
0.726) 2px, transparent 30px),
    radial-gradient(rgb(59, 161, 230), rgb(34, 141,
212) ,#3049d6, #0820ac,#06156e ,#040d42,#02092e);
  background-size: 350px 350px, 300px 300px, 250px 250px,
200px 200px, 160px 160px,cover;
  background-position: center;
  --page-bg: #f5f5f5;
  --dark-text: #2A2935;
  --baseline: 1.2vw;
  --base-size: var(--baseline) * 1.2;
}
* {
  box-sizing: border-box;
}
.corp {
  justify-content: center;
}
.articol-cover {
  position: relative;
  top: 8vw;
  width: calc(var(--

```



```

baseline) * 60);
  height: calc(var(--baseline) * 25);
  box-shadow: 0 0 100px rgb(0, 0, 0);
}
.book {
  width: 100%;
  height: 100%;
  display: flex;
  perspective: 1200px;

  &__page {
    position: relative;
    width: 50%;
    height: 100%;
    display: grid;
    transform: rotateY(0deg);
    transition: transform 0.9s cubic-bezier(0.645, 0.045, 0.355,
1);
    transform-origin: 0% 0%;
    background-color: var(--page-bg);
    background-image: linear-gradient(90deg, rgb(199, 197, 197)
0%, rgba(247,247,247,0) 18%);
    &:nth-of-type(1) {
      background-image: linear-gradient(-90deg, rgb(199, 197, 197)
0%, rgba(10, 10, 10, 0) 18%);
    }
    &--1 {
      cursor: pointer;
      overflow: hidden;
    }
    &--2 {
      position: absolute;
      right: 0;
      pointer-events: none;
      transform-style: preserve-3d;
      background-color: var(--page-bg);
      background-image: linear-gradient(90deg, rgb(199, 197, 197)
0%, rgba(247,247,247,0) 18%);
    }
    &--4 {
      cursor: pointer;
      padding: 0 calc(var(--baseline) * 3);
    }
    &-front {
      position: absolute;
      width: 100%;
      height: 100%;
      transform: rotateY(0deg) translateZ(1px);
    }
    &-back {
      position: absolute;
      width:
100%;

```



```

        height: 100%;
        padding: 0 calc(var(--baseline) * 1.8);
        transform: rotateY(180deg) translateZ(1px);
    }
}
input[type="radio"] {
    display: none;

    &:checked+.book__page {
        transition: transform 0.9s cubic-bezier(0.645, 0.045, 0.355,
1);
        transform: rotateY(-180deg);
    }
}
}
.titluart1{
    position: relative;
    font-family:"Lucida Console", Courier , monospace ;
    font-size: 3vw;
    top: 10vw;
    left: 5vw;
}
.text{
    position: relative;
    font-family:"Lucida Console", Courier , monospace ;
    font-size: 1.5vw;
    top: 1.5vw;
    left: 2vw;
}
.autorart1{
    position: relative;
    font-family:"Lucida Console", Courier , monospace ;
    top: 5vw;
    left: 15vw;
}
.dataart1{
    position: relative;
    font-family:"Lucida Console", Courier , monospace ;
    top: 5vw;
    left: 2.5vw;
}
@media only screen and (max-width: 1200px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 30);
        top: 10vw;
    }
}
@media only screen and (max-width: 1100px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height:
            calc(var(--

```





```

baseline) * 35);
    top: 10vw;
  }
}
@media only screen and (max-width: 1000px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 40);
    top: 10vw;
  }
}
@media only screen and (max-width: 900px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 45);
    top: 10vw;
  }
}
@media only screen and (max-width: 855px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 37);
    top: 15vw;
  }
}
@media only screen and (max-width: 800px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 40);
    top: 15vw;
  }
}
@media only screen and (max-width: 700px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 45);
    top: 15vw;
  }
}
@media only screen and (max-width: 640px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 50);
    top: 15vw;
  }
}
@media only screen and (max-width: 600px){

  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 70);
    top: 20vw;
  }
}

```



```

    }
  }
  @media only screen and (max-width: 550px){

    .articol-cover {
      width: calc(var(--baseline) * 70);
      height: calc(var(--baseline) * 75);
      top: 20vw;
    }
  }
  @media only screen and (max-width: 520px){
    .articol-cover {
      width: calc(var(--baseline) * 70);
      height: calc(var(--baseline) * 80);
      top: 20vw;
    }
  }
  @media only screen and (max-width: 470px){

    .articol-cover {
      width: calc(var(--baseline) * 70);
      height: calc(var(--baseline) * 90);
      top: 20vw;
    }
  }
}

```

### **SecondArticol.tsx**

```

import React from "react";
import {useState,useEffect} from 'react';
import "../SecondArticol.scss"
import data from "../jSecondArticol1.json"
export const SecondArticol = () =>{
  return(
    <div className="corp">
      {data.map((postData) => {
        console.log(postData);
        return (
          <div className="articol-cover" >
            <div className="book">
              <label htmlFor="page-1" className="book__page book__page--1">
                <div className="titluart1">{postData.titluart1}</div>
                <div className="autorart1">{postData.autorart1}</div>
                <div className="dataart1">{postData.dataart1}</div>
              </label>
              <label htmlFor="page-2" className="book__page book__page--4">
                <div className="content">
                  <div className="text">
                    {postData.pagina3}
                  </div>
                </div>
              </label>
            </div>
          </div>
        )
      })}
    </div>
  )
}

```



```

</label>
<input type="radio" name="page" id="page-1"/>
<input type="radio" name="page" id="page-2"/>
<label className="book__page book__page--2">
<div className="book__page-front">
  <div className="content">
    <div className="text">{postData.pagina1}</div>
  </div>
</div>
<div className="book__page-back">
  <div className="content">
    <div className="text">{postData.pagina2}</div>
  </div>
</div>
</label>
</div>
</div>
);
}}
</div>
)
}

```

### JSecondArticol1.json

```

[
  { "titluart1": "Titlul titlul titlul titlul",
    "autorart1": "Autor1",
    "dataart1": "01.01.2021",
    "pagina1": "Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1
Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1
Pagina1 Pagina1 Pagina1 Pagina1",
    "pagina2": "Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2
Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2
Pagina2 Pagina2 Pagina2 Pagina2",
    "pagina3": "Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3
Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3
Pagina3 Pagina3 Pagina3 Pagina3 "
  }
]

```

### ThirdArticol.scss

```

.corp {
  width: 100%;
  height: 100%;
  display: flex;
  margin: 0;
  padding: 1rem;
  min-height: 100vh;
}

```



```

background: #a2d4ff;
background-attachment: fixed;
background-image:
    radial-gradient(rgb(19, 170, 216), rgba(57, 86, 250, 0.767)
2px, transparent 40px),
    radial-gradient(rgb(19, 170, 216), rgba(71, 124, 238, 0.904)
1px, transparent 30px),
    radial-gradient(rgb(19, 170, 216), rgba(77, 195, 250, 0.849)
2px, transparent 40px),
    radial-gradient(rgba(19, 170, 216,.4), rgba(25, 102, 202,
0.726) 2px, transparent 30px),
    radial-gradient(rgba(19, 170, 216,.4), rgba(36, 141, 211,
0.726) 2px, transparent 30px),
    radial-gradient(rgb(59, 161, 230), rgb(34, 141,
212) ,#3049d6, #0820ac,#06156e ,#040d42,#02092e);
background-size: 350px 350px, 300px 300px, 250px 250px,
200px 200px, 160px 160px,cover;
background-position: center;
--page-bg: #f5f5f5;
--dark-text: #2A2935;
--baseline: 1.2vw;
--base-size: var(--baseline) * 1.2;
}
* {
    box-sizing: border-box;
}
.corp {
    justify-content: center;
}
.articol-cover {
    position: relative;
    top: 8vw;
    width: calc(var(--baseline) * 60);
    height: calc(var(--baseline) * 25);
    box-shadow: 0 0 100px rgb(0, 0, 0);
}
.book {
    width: 100%;
    height: 100%;
    display: flex;
    perspective: 1200px;
    &__page {
        position: relative;
        width: 50%;
        height: 100%;
        display: grid;
        transform: rotateY(0deg);
        transition: transform 0.9s cubic-bezier(0.645, 0.045, 0.355,
1);
        transform-origin: 0% 0%;
        background-color: var(--page-bg);
        background-
            image: linear-

```



```

gradient(90deg, rgb(199, 197, 197) 0%, rgba(247,247,247,0) 18%);
  &:nth-of-type(1) {
    background-image: linear-gradient(-90deg, rgb(199, 197, 197)
0%, rgba(10, 10, 10, 0) 18%);
  }
  &--1 {
    cursor: pointer;
    overflow: hidden;
  }
  &--2 {
    position: absolute;
    right: 0;
    pointer-events: none;
    transform-style: preserve-3d;
    background-color: var(--page-bg);
    background-image: linear-gradient(90deg, rgb(199, 197, 197)
0%, rgba(247,247,247,0) 18%);

  }
  &--4 {
    cursor: pointer;
    padding: 0 calc(var(--baseline) * 3);
  }
  &-front {
    position: absolute;
    width: 100%;
    height: 100%;
    transform: rotateY(0deg) translateZ(1px);
  }
  &-back {
    position: absolute;
    width: 100%;
    height: 100%;
    padding: 0 calc(var(--baseline) * 1.8);
    transform: rotateY(180deg) translateZ(1px);
  }
}
input[type="radio"] {
  display: none;
  &:checked+.book__page {
    transition: transform 0.9s cubic-bezier(0.645, 0.045, 0.355,
1);
    transform: rotateY(-180deg);
  }
}
}
.titluart1{
  position: relative;
  font-family:"Lucida Console", Courier , monospace ;
  font-size: 3vw;
  top: 10vw;
  left: 5vw;

```



```

}
.text{
  position: relative;
  font-family:"Lucida Console", Courier , monospace ;
  font-size: 1.5vw;
  top: 1.5vw;
  left: 2vw;
}
.autorart1{
  position: relative;
  font-family:"Lucida Console", Courier , monospace ;
  top: 5vw;
  left: 15vw;
}
.dataart1{
  position: relative;
  font-family:"Lucida Console", Courier , monospace ;
  top: 5vw;
  left: 2.5vw;
}
@media only screen and (max-width: 1200px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 30);
    top: 10vw;
  }
}
@media only screen and (max-width: 1100px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 35);
    top: 10vw;
  }
}
@media only screen and (max-width: 1000px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 40);
    top: 10vw;
  }
}
@media only screen and (max-width: 900px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--baseline) * 45);
    top: 10vw;
  }
}
@media only screen and (max-width: 855px){
  .articol-cover {
    width: calc(var(--baseline) * 70);
    height: calc(var(--

```



```

baseline) * 37);
    top: 15vw;
}
}
@media only screen and (max-width: 800px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 40);
        top: 15vw;
    }
}
@media only screen and (max-width: 700px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 45);
        top: 15vw;
    }
}
@media only screen and (max-width: 640px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 50);
        top: 15vw;
    }
}
@media only screen and (max-width: 600px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 70);
        top: 20vw;
    }
}

@media only screen and (max-width: 550px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 75);
        top: 20vw;
    }
}
@media only screen and (max-width: 520px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 80);
        top: 20vw;
    }
}
@media only screen and (max-width: 470px){
    .articol-cover {
        width: calc(var(--baseline) * 70);
        height: calc(var(--baseline) * 90);
        top: 20vw;
    }
}

```



```

    }
  }
}

```

### ThirdArticol.tsx

```

import React from "react";
import {useState,useEffect} from 'react';
import "../ThirdArticol.scss"
import data from "../jThirdArticol1.json"
export const ThirdArticol = () =>{
  return(
    <div className="corp">
      {data.map((postData) => {
        console.log(postData);
        return (
          <div className="articol-cover" >
            <div className="book">
              <label htmlFor="page-1" className="book__page book__page--1">
                <div className="titluart1">{postData.titluart1}</div>
                <div className="autorart1">{postData.autorart1}</div>
                <div className="dataart1">{postData.dataart1}</div>
              </label>
              <label htmlFor="page-2" className="book__page book__page--4">
                <div className="content">
                  <div className="text">
                    {postData.pagina3}
                  </div>
                </div>
              </label>
              <input type="radio" name="page" id="page-1"/>
              <input type="radio" name="page" id="page-2"/>
              <label className="book__page book__page--2">
                <div className="book__page-front">
                  <div className="content">
                    <div className="text">{postData.pagina1}</div>
                  </div>
                </div>
                <div className="book__page-back">
                  <div className="content">
                    <div className="text">{postData.pagina2}</div>
                  </div>
                </div>
              </label>
            </div>
          </div>
        )
      })}
    </div>
  )
}

```





### jThirdArticol.json

```
[
  { "titluart1":"Titlul titlul titlul titlul",
    "autorart1":"Autor1",
    "dataart1":"01.01.2021",
    "pagina1":"Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1
Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1 Pagina1
Pagina1 Pagina1 Pagina1 Pagina1",
    "pagina2":"Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2
Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2 Pagina2
Pagina2 Pagina2 Pagina2 Pagina2",
    "pagina3":"Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3
Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3 Pagina3
Pagina3 Pagina3 Pagina3 Pagina3 "
  }
]
```

### SideBar.scss

```
/* Position and sizing of burger button */
.bm-burger-button {
  position: fixed;
  width: 36px;
  height: 30px;
  left: 36px;
  top: 36px;
}
/* Color/shape of burger icon bars */
.bm-burger-bars {
  background: #373a47;
}
/* Color/shape of burger icon bars on hover*/
.bm-burger-bars-hover {
  background: #a90000;
}
/* Position and sizing of clickable cross button */
.bm-cross-button {
  height: 24px;
  width: 24px;
}
/* Color/shape of close button cross */
.bm-cross {
  background: #bdc3c7;
}
/*
Sidebar wrapper styles
Note: Beware of modifying this element as it can break the
animations - you should not need to touch it in most cases
*/
```



```

.bm-menu-wrap {
  position: fixed;
  height: 100%;
}
/* General sidebar styles */
.bm-menu {
  background: #373a47;
  padding: 2.5em 1.5em 0;
  font-size: 1.15em;
}
/* Morph shape necessary with bubble or elastic */
.bm-morph-shape {
  fill: #373a47;
}
/* Wrapper for item list */
.bm-item-list {
  color: #b8b7ad;
  padding: 0.8em;
}
/* Individual item */
.bm-item {
  display: inline-block;
}
/* Styling of overlay */
.bm-overlay {
  background: rgba(0, 0, 0, 0.3);
}

.ShowSideBarIcon {
  position: fixed;
  float: left;
  margin-left: 30px;
  z-index: 100 !important;
  font-size: 2rem;
  height: 80px;
  display: flex;
  justify-content: flex-start;
  align-items: center;
  color: white;
}
.slideBarData {
  margin-top: 100px;
  z-index: 998;
}

```

### **SideBar.tsx**

```

import React, { useState } from "react";
import styled from "styled-components";
import { Link } from "react-router-

```



```

dom";
import { SidebarData } from "../SideBarData.js";
import SubMenu from "../SubMenu";
import { Menu } from "react-feather";
import "../SideBar.scss";
import LanguageSelector from "../LanguageSelector";
import { useTranslation } from "react-i18next";
import "../../services/i18n";
const SidebarNav = styled.nav`
  background: #4E5478;
  width: 200px;
  height: 100vh;
  margin: 100px;
  display: flex;
  justify-content: center;
  position: fixed;
  top: 60px;
  left: ${({ sidebar }) => (sidebar ? "0" : "-100%")};
  transition: 490ms;
  order: 2
`;
const SidebarWrap = styled.div`
  width: 100%;
  margin-top: 10px;
  // padding: 0 100px;
`;
const Sidebar = () => {
  const [sidebar, setSidebar] = useState(false);
  const showSidebar = () => setSidebar(!sidebar);
  const { t, i18n } = useTranslation();
  const changeLanguage = (event) => {
    i18n.changeLanguage(event.target.value);
  };
  return (
    <>
      <Menu className="ShowSideBarIcon" onClick={showSidebar} />
      <SidebarNav sidebar={sidebar} >
        <SidebarWrap>
          <div className="slideBarData">
            {SidebarData.map((item, index) => {
              return <SubMenu item={item} key={index}
showSidebar={showSidebar}/>;
            })}
          </div>
          <LanguageSelector />
        </SidebarWrap>
      </SidebarNav>
    </>
  );
};
export default Sidebar;

```



## SideBarData.js

```
export const SidebarData = [
  {
    title: "Students",
    path: "#",
    subNav: [
      {
        title: "Projects",
        path: "/about-us/aim",
      },
    ],
  },
  {
    title: "Admin",
    path: "#",
    restriction: 'admin',
    subNav: [
      {
        title: "Database",
        path: "/admin/database",
        restriction: 'admin2',
        cName: "sub-nav",
      },
      {
        title: "Service 2",
        path: "/services/services2",
        restriction: 'admin',
        cName: "sub-nav",
      },
      {
        title: "Service 3",
        path: "/services/services3",
        restriction: 'admin',
      },
    ],
  },
  {
    title: "Services",
    path: "/services",
  },
  {
    title: "Projects",
    path: "#",
    restriction: 'admin',
    subNav: [
      {
        title: "List",
        restriction: 'admin',
        path: "/projects/list",
      },
      {
        title: "Attendace",
```



```

        restriction: 'admin',
        path: "/events/attendance",
      },
      {
        title: "Event 2",
        path: "/events/events2",
      },
    ],
  },
  {
    title: "Support",
    path: "/support",
  },
  {
    title: "Contact",
    path: "/contact",
  },
];

```

### SubMenu.tsx

```

import React, { useState } from "react";
import { Link } from "react-router-dom";
import styled from "styled-components";
const SidebarLink = styled(Link)`
  display: flex;
  color: #ele9fc;
  justify-content: space-between;
  align-items: center;
  padding: 20px;
  list-style: none;
  height: 60px;
  text-decoration: none;
  font-size: 18px;
  &:hover {
    background: #252831;
    border-left: 4px solid blue;
    cursor: pointer;
  }
`;
const SidebarLabel = styled.span`
  margin-left: 16px;
`;
const DropdownLink = styled(Link)`
  background: #4E5478;
  height: 60px;
  padding-left: 3rem;
  display: flex;
  align-items: center;
  text-decoration: none;

```



```

    color: white;
    font-size: 18px;
    &:hover {
      background: #252831;
      cursor: pointer;
      border-left: 6px solid blue;
    }
  `;
const SubMenu = ({ item, showSidebar }) => {
  const [subnav, setSubnav] = useState(false);
  const showSubnav = () => setSubnav(!subnav);
  return (
    <>
      {item.restriction === 'admin' ?
        <SidebarLink to={item.path} onClick={item.path === "#" ?
item.subNav && showSubnav : showSidebar }>
          <div>
            {item.icon}
            <SidebarLabel>{item.title}</SidebarLabel>
          </div>
          <div>
            {item.subNav && subnav && showSidebar
              ? item.iconOpened
              : item.subNav
              ? item.iconClosed
              : null}
          </div>
        </SidebarLink>
        : null }
      {subnav &&
        item.subNav.map((item, index) => {
          return (
            item.restriction === 'admin' ?
              <DropDownLink to={item.path} key={index}
onClick={showSidebar}>
                {item.icon}
                <SidebarLabel>{item.title}</SidebarLabel>
              </DropDownLink>
              : null
            );
          )}}
    </>
  );
};
export default SubMenu;

```

### ToggleSwitch.scss

```

.switch {
  position: absolute;
  display: inline-block;

```



```

    left:50px;
    right:auto;
    bottom: 100px;
}
.switch > span {
  position: absolute;
  top: 10px;
  pointer-events: none;
  font-family: 'Helvetica', Arial, sans-serif;
  font-weight: bold;
  font-size: 12px;
  text-transform: uppercase;
  text-shadow: 0 1px 0 rgba(0, 0, 0, .06);
  width: 50%;
  text-align: center;
}
input.check-toggle-round-flat:checked ~ .off {
  color: #373a47;
}
input.check-toggle-round-flat:checked ~ .on {
  color: #fff;
}
.switch > span.on {
  left: 0;
  padding-left: 2px;
  color: #373a47;
}
.switch > span.off {
  right: 0;
  padding-right: 4px;
  color: #fff;
}
.check-toggle {
  position: absolute;
  margin-left: -9999px;
  visibility: hidden;
}
.check-toggle + label {
  display: block;
  position: relative;
  cursor: pointer;
  outline: none;
  -webkit-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  user-select: none;
}
input.check-toggle-round-flat + label {
  padding: 2px;
  width: 97px;
  height: 35px;
  background-
    color: #3f51b5;

```



```

    -webkit-border-radius: 60px;
    -moz-border-radius: 60px;
    -ms-border-radius: 60px;
    -o-border-radius: 60px;
    border-radius: 60px;
}
input.check-toggle-round-flat + label:before, input.check-toggle-
round-flat + label:after {
    display: block;
    position: absolute;
    content: "";
}
input.check-toggle-round-flat + label:before {
    top: 2px;
    left: 2px;
    bottom: 2px;
    right: 2px;
    background-color: #3f51b5;
    -moz-border-radius: 60px;
    -ms-border-radius: 60px;
    -o-border-radius: 60px;
    border-radius: 60px;
}
input.check-toggle-round-flat + label:after {
    top: 4px;
    left: 4px;
    bottom: 4px;
    width: 48px;
    background-color: #fff;
    -webkit-border-radius: 52px;
    -moz-border-radius: 52px;
    -ms-border-radius: 52px;
    -o-border-radius: 52px;
    border-radius: 52px;
    -webkit-transition: margin 0.2s;
    -moz-transition: margin 0.2s;
    -o-transition: margin 0.2s;
    transition: margin 0.2s;
}
input.check-toggle-round-flat:checked + label {
}
input.check-toggle-round-flat:checked + label:after {
    margin-left: 44px;}

```

### **ToggleSwitch.tsx**

```

import React from "react";
import PropTypes from "prop-types";
import "../ToggleSwitch.scss";
/*
Toggle Switch                                Component

```





Note: id, checked and onChange are required for ToggleSwitch component to function. The props name, small, disabled and optionLabels are optional.

Usage: <ToggleSwitch id="id" checked={value} onChange={checked => setValue(checked)} />

```
*/
const ToggleSwitch = ({
  id,
  name,
  checked,
  onChange,
  optionLabels,
  small,
  disabled,
}) => {
  function handleKeyPress(e) {
    if (e.keyCode !== 32) return;
    e.preventDefault();
    onChange(!checked);
  }
  return (
    <div className="switch">
      <input
        type="checkbox"
        name={name}
        className="check-toggle check-toggle-round-flat"
        id={id}
        checked={checked}
        onChange={e => onChange(e.target.checked)}
        disabled={disabled}
      />
      <label
        className="language-toggle"
        tabIndex={disabled ? -1 : 1}
        onKeyDown={e => handleKeyPress(e)}
        htmlFor={id}
      > </label>
      <span className="on">RO</span>
      <span className="off">EN</span>
    </div>
  );
};

ToggleSwitch.defaultProps = {
  optionLabels: ["EN", "RO"],
};

ToggleSwitch.propTypes = {
  id: PropTypes.string.isRequired,
  checked: PropTypes.bool.isRequired,
  onChange: PropTypes.func.isRequired,
  name: PropTypes.string,
  optionLabels: PropTypes.array,
  small: PropTypes.bool,
```



```

    disabled: PropTypes.bool,
  };
export default ToggleSwitch;

```

### **LanguageSelector.tsx**

```

import React, { useState, useEffect } from "react";
import { useTranslation } from "react-i18next";
import ToggleSwitch from "../toggleSwitch/ToggleSwitch";
const LanguageSelector = () => {
  const { t, i18n } = useTranslation();
  let [lang, setLang] = useState(true);
  useEffect(() => {
    i18n.changeLanguage("ro");
    setLang(false);
  }, []);
  const changeLanguage = (checked) => {
    setLang(checked);
    if (checked) {
      i18n.changeLanguage("en");
    }
    if (!checked) {
      i18n.changeLanguage("ro");
    }
  };
  return (
    <div>
      <ToggleSwitch id="language" checked={lang}
onChange={changeLanguage} />
    </div>
  );
};
export default LanguageSelector;

```

### **UseMemo.tsx**

```

import React, { useState } from 'react';
export const UseMemo = () => {
  const [number, setNumber] = useState(1);
  const [inc, setInc] = useState(0);
  const factorial = factorialOf(number);
  const onChange = event => {
    setNumber(Number(event.target.value));
  };
  const onClick = () => setInc(i => i + 1);
  return (
    <div style={{ marginTop: `200px` }}>
      Factorial of
      <input type="number" value={number} onChange={onChange} />
      is {factorial}
    </div>
  );
};

```



```

        <button onClick={onClick}>Re-render</button>
      </div>
    );
  }
  function factorialOf(n) {
    console.log('factorialOf(n) called!');
    return n <= 0 ? 1 : n * factorialOf(n - 1);
  }

```

### **genericModal.scss**

```

.generic-modal {
  border: 2px solid red;
  align-items: center;
}
.container{
  display: flex;
  align-items: center;
}
.border{
  border-bottom: 1px solid black;
  width: 100%;
}
.content {
  padding: 0 10px 0 10px;
}

```

### **genericModal.tsx**

```

import React from 'react';
import { makeStyles, Theme, createStyles } from '@material-ui/
core/styles';
import Modal from '@material-ui/core/Modal';
import CloseIcon from '@material-ui/icons/Close';
import './genericModal.scss'
const useStyles = makeStyles((theme: Theme) =>
  createStyles({
    paper: {
      position: 'absolute',
      width: 500,
      backgroundColor: theme.palette.background.paper,
      border: '2px solid #000',
      boxShadow: theme.shadows[5],
      padding: theme.spacing(2, 4, 3),
    },
    button: {
      float: 'right',
      backgroundColor: 'red',
    },
  })),

```



```

);
export default function GenericModal(props: any) {
  const classes = useStyles();
  const [modalStyle] = React.useState(props.getModalStyle);
  return (
    <div>
      <Modal
        className="generic-modal"
        body={props.body}
        open={props.open}
        onClose={props.handleModal}
      >
        <div style={modalStyle} className={classes.paper}>
          <div><b>{props.title}</b>
            <button className={classes.button} type="button"
onClick={props.handleModal}>
              <CloseIcon />
            </button></div>
          <div>{props.body}</div>
        </div>
      </Modal>
    </div>
  );
}

```

### **authGuard.js**

```

import React, { Component } from 'react'
import { connect } from 'react-redux'
import Axios from 'axios'
import { config } from ".././config";
export default OriginalComponent => {
  class MixedComponent extends Component {
    checkAuth() {
      var currentntDate = new Date()
      if (!sessionStorage.getItem('exp') || !
sessionStorage.getItem('token') || currentntDate > new
Date(sessionStorage.getItem('exp'))) {
        sessionStorage.removeItem('exp')
        sessionStorage.removeItem('userEmail')
        sessionStorage.removeItem('name')
        sessionStorage.removeItem('token')
        sessionStorage.removeItem('roles')
        window.open(config.baseURL + config.baseLOCATION + "/"
auth/logout", "_self");
      }
    }
    componentDidMount() {
      this.checkAuth()
    }
  }
}

```



```

componentDidUpdate() {
  this.checkAuth()
}
render() {
  return <OriginalComponent {...this.props} />
}
}
function MapStateToProps(state) {
  return {
    isAuthenticated: state.auth.isAuthenticated,
  }
}
return connect(MapStateToProps)(MixedComponent)
}

```

### **utils.tsx**

```

import React, {useEffect, useState} from "react";
export function useMediaQuery(query) {
  const [matches, setMatches] = useState(false);
  useEffect(() => {
    const media = window.matchMedia(query);
    if (media.matches !== matches) {
      setMatches(media.matches);
    }
    const listener = () => {
      setMatches(media.matches);
    };
    media.addListener(listener);
    return () => media.removeListener(listener);
  }, [matches, query]);
  return matches;
}
export const useIsSmall = () => useMediaQuery('(min-width: 480px)');
export const useIsMedium = () => useMediaQuery('(min-width: 768px)');

```

### **LogOut.tsx**

```

import React, { useState } from "react";
import { config } from "../config"
function LogOut() {
  window.open(config.baseURL + config.baseLOCATION + "/auth/logout", "_self");
  sessionStorage.removeItem('exp')
  sessionStorage.removeItem('userEmail')
  sessionStorage.removeItem('name')
  sessionStorage.removeItem('token')
  sessionStorage.removeItem('roles')
}

```



```
export default LogOut
```

### **auth.js**

```
import {
  AUTH_SIGN_UP,
  AUTH_SIGN_IN,
  AUTH_SIGN_OUT,
  AUTH_ERROR
} from "../actions/types";
const DEFAULT_STATE = {
  isAuthenticated: false,
  name: "",
  role: "",
  token: "",
  errorMessage: "",
  nokiaid: ""
};
export default (state = DEFAULT_STATE, action) => {
  switch (action.type) {
    case AUTH_SIGN_UP:
      return {
        ...state,
        token: action.payload,
        name: action.payload_name,
        email: action.payload_email,
        role: action.payload_role,
        nokiaid: action.payload_nokiaid,
        isAuthenticated: true,
        errorMessage: ""
      };
    case AUTH_SIGN_IN:
      return {
        ...state,
        token: action.payload,
        name: action.payload_name,
        email: action.payload_email,
        role: action.payload_role,
        nokiaid: action.payload_nokiaid,
        isAuthenticated: true,
        errorMessage: ""
      };
    case AUTH_SIGN_OUT:
      return {
        ...state,
        token: action.payload,
        name: action.payload_name,
        email: action.payload_email,
        role: action.payload_role,
        nokiaid: action.payload_nokiaid,
        isAuthenticated:

```



```

false,
  errorMessage: ""
};
case AUTH_ERROR:
  return { ...state, errorMessage: action.payload };
default:
  return state;
}
};

```

### **index.js**

```

import { combineReducers } from 'redux';
import { reducer as formReducer } from 'redux-form';
import authReducer from './auth';
export default combineReducers({
  form: formReducer,
  profile: profileReducer,
  auth: authReducer
});

```

### **enableButton.js**

```

export const enableButton = (index) => {
  return {
    type: 'ENABLE BUTTON',
    index: index
  }
}

```

### **getProjects.js**

```

export const getProjects = (data) => {
  return {
    type: 'GET PROJECTS',
    payload: {
      data: data
    }
  }
}

```

### **reviewTask.js**

```

export const reviewTask = (projectIndex, taskIndex) => {
  return {
    type: 'REVIEW TASK',
    payload: {
      projectIndex:

```



```
projectIndex,  
      taskIndex: taskIndex,  
    },  
  }  
}
```

### **setDefaultValue.js**

```
export const setDefaultValue = (index, value) => {  
  return {  
    type: 'SET DEFAULT VALUE',  
    index: index,  
    value: value  
  }  
}
```

### **setTaskCompetenceLevel.js**

```
export const setTaskCompetenceLevel = (projectIndex, taskIndex,  
value) => {  
  return {  
    type: 'SET TASK COMPETENCE LEVEL',  
    value: value,  
    projectIndex: projectIndex,  
    taskIndex: taskIndex  
  }  
}
```

### **toggleDisplay.js**

```
export const toggleDisplay = (index) => {  
  return {  
    type: 'TOGGLE DISPLAY',  
    payload: index}}}
```

### **getTasks.js**

```
export const getTasks = (data) => {  
  return {  
    type: 'GET TASKS',  
    payload: {  
      data: data  
    }  
  }  
}
```





### onDeleteTask.js

```
export const onDeleteTask = (index) => {
  return {
    type: 'DELETE',
    payload: {
      index: index
    }
  }
}
```

### setTask.js

```
export const setTask = (index, id, data) => {
  return {
    type: 'SET TASK',
    payload: {
      index: index,
      id: id,
      data: data
    }
  }
}
```

### auth.js

```
import { UPDATE_PROFILE_TYPE, AUTH_SIGN_IN, AUTH_SIGN_OUT,
AUTH_ERROR } from "../types";
const DEFAULT_STATE = {
  isAuthenticated: false,
  role: "",
  errorMessage: "",
};
export default (state = DEFAULT_STATE, action) => {
  switch (action.type) {
    case UPDATE_PROFILE_TYPE:
      return {
        ...state,
        type: action.payload.type
      };
    case AUTH_SIGN_IN:
      return {
        ...state,
        token: action.payload.token,
        name: action.payload.name,
        email: action.payload.email,
        role: action.payload.role,
        nokiaid: action.payload.nokiaid,
        isAuthenticated:

```



```

true,
  errorMessage: "",
};
case AUTH_SIGN_OUT:
  return {
    ...state,
    access_token: action.payload,
    username: action.payload_username,
    role: action.payload_role,
    isAuthenticated: false,
    errorMessage: "",
  };
case AUTH_ERROR:
  return { ...state, errorMessage: action.payload };
default:
  return state;}};

```

### **counter.js**

```

const counterReducer = (state = 0, action) => {
  switch (action.type) {
    case "INCREMENT":
      return state + 1;
    case "DECREMENT":
      return state - 1;
    default: {
      return state;
    }
  }
};
export default counterReducer;

```

### **index.js**

```

import { combineReducers } from "redux";
import authReducer from "./auth";
import projectsReducer from './projects'
import tasksReducer from './tasks';
export default combineReducers({
  auth: authReducer,
  projectsReducer,
  tasksReducer,
});

```

### **islogged.js**

```

const loggedReducer = (state = false, action) => {
  switch (action.type) {
    case "SIGN_IN":

```



```

        return !state;
    default:
        return false;
    }
};
export default loggedReducer;
projects.js

```

```

const initialState = {
  projects: []
};
const projectsReducer = (state = initialState, action) => {
  switch (action.type) {
    case 'TOGGLE DISPLAY': return ({
      ...state,
      projects: state.projects.map((project, index) => {
        if (index !== action.payload) {
          return project;
        }
        return {
          ...project,
          display: !project.display
        }
      })
    });
    case 'ENABLE BUTTON': return ({
      ...state,
      projects: state.projects.map((project, index) => {
        if (index !== action.index) {
          return project;
        }
        return {
          ...project,
          buttonDisabled: false
        }
      })
    });
    case 'SET DEFAULT VALUE': return ({
      ...state,
      projects: state.projects.map((project, index) => {
        if (index !== action.index) {
          return project;
        }
        return (
          {
            ...project,
            tsks: project.tsks.map(task => {
              return (
                {
                  ...task,
                  competenceLevel: action.value,

```



```

        }
      )
    })),
    verifiedTasks: 0,
  }
);
});
case 'SET TASK COMPETENCE LEVEL': return ({
  ...state,
  projects: state.projects.map((project, index) => {
    if (index !== action.projectIndex) {
      return project;
    }
    return (
      {
        ...project,
        tsks: project.tsks.map((task, index) => {
          if (index !== action.taskIndex)
            return task;
          return (
            {
              ...task,
              competenceLevel: action.value,
            }
          )
        })
      }
    );
  })
);
});
case 'REVIEW TASK': return ({
  ...state,
  projects: state.projects.map((project, pIndex) => {
    if (pIndex !== action.payload.projectIndex) {
      return project;
    }
    let ok;
    return (
      {
        ...project,
        tsks: project.tsks.map((task, tIndex) => {
          if (tIndex !==
action.payload.taskIndex) {
            return task;
          }
          ok = !task.reviewed;
          return ({
            ...task,
            reviewed: !task.reviewed,
          });
        })
      }
    );
  })
);

```



```

        }),
        verifiedTasks: ok ? project.verifiedTasks
+ 1 : project.verifiedTasks - 1,
    }
  );
  })
});
case 'GET PROJECTS': return({
  ...state,
  projects: action.payload.data,
});
default: return { ...state };
}
}
export default projectsReducer;

```

### **tasks.js**

```

import axios from 'axios'
import { config } from "../../config"
const initialState = {
  tasks: []
};
const tasksReducer = (state = initialState, action) => {
  switch (action.type) {
    case 'DELETE':
      const tasks = {
        ...state,
        tasks: state.tasks.filter((task, id) => task.id !==
action.payload.index),
      }
      axios.delete(config.baseURL + config.baseLOCATION + `/
dailyTasks/${action.payload.index}`)
        .catch(err => {
          console.log(err);
        });
      return tasks;
    case 'SET TASK':
      axios.put(config.baseURL + config.baseLOCATION + `/
dailyTasks/${action.payload.id}`, action.payload.data)
        .catch(err => {
          console.log(err);
        });
      return ({
        ...state,
        tasks: state.tasks.map((task, index) => {
          if (index !== action.payload.index) {
            return task;
          }
          return action.payload.data;
        })
      });
  }
}

```



```

    });
    case 'GET TASKS': return ({
      ...state,
      tasks: action.payload.data,
    });
    default: return { ...state };
  }
}
export default tasksReducer;

```

### types.js

```

export const UPDATE_PROFILE_TYPE = "UPDATE_PROFILE_TYPE";
export const AUTH_SIGN_IN = "AUTH_SIGN_IN";
export const AUTH_SIGN_OUT = "AUTH_SIGN_OUT";
export const AUTH_ERROR = "AUTH_ERROR";

```

### i18n.js

```

import i18n from "i18next";
import Backend from "i18next-xhr-backend";
import { initReactI18next } from "react-i18next";
// process.env.NODE_ENV == "development" ? location = "." :
location = "/public"
console.log('ENV', process.env.NODE_ENV)
i18n
  .use(Backend)
  .use(initReactI18next)
  .init({
    lng: "en",
    ns: ["translations"],
    backend: {
      /* translation file path */
      loadPath: `./assets/i18n/${ns}/${lng}.json`,
    },
    fallbackLng: "en",
    debug: true,
    /* can have multiple namespace, in case you want to divide a
    huge translation into smaller pieces and load them on demand */
    defaultNS: "translations",
    keySeparator: false,
    interpolation: {
      escapeValue: false,
      formatSeparator: ",",
    },
    react: {
      useSuspense: true,
    },
  });
export default i18n;

```



### App.js

```
import React, { Component, Suspense } from "react";
import { AppRouter } from "../AppRouter";
class App extends Component {
  render() {
    return <Suspense fallback="loading"><AppRouter /></Suspense>;
  }
}
export default App;
```

### App.test.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import App from '../App';
it('renders without crashing', () => {
  const div = document.createElement('div');
  ReactDOM.render(<App />, div);
  ReactDOM.unmountComponentAtNode(div);
});
```

### AppRouter.js

```
import React from "react";
import Landingpage from "../components/LandingPage/Landingpage.jsx";
import LoginError from "../components/common/LoginError/LoginError";
import Homepage from "../components/Homepage/Homepage";
import Footer from "../components/Footer/Footer";
import { Header } from "../components/Header/Header";
import { HashRouter, Route } from "react-router-dom";
import { UseMemo } from "../components/UseMemo";
import { Credits } from "../components/Credits/Credits";
import { Newsletter } from "../components/Newsletter/Newsletter";
import { FirstArticol } from "../components/Newsletter/FirstArticol/FirstArticol";
import { SecondArticol } from "../components/Newsletter/SecondArticol/SecondArticol";
import { ThirdArticol } from "../components/Newsletter/ThirdArticol/ThirdArticol";
export const AppRouter = (props) => {
  return (
    <HashRouter {...props} >
      <Route exact path={"/"} component={Landingpage} />
      <Route exact path={"/error"} component={LoginError} />
      <Route exact path={"/home"} component={Homepage} />
      <Route
        exact path={"/" />

```



```

usememo"} component={UseMemo} />
    <Route exact path={"/credits"} component={Credits} />
    <Route exact path={"/newsletter"} component={Newsletter} />
    <Route exact path={"/firstarticol"} component={FirstArticol}
/>
    <Route exact path={"/secondarticol2"}
component={SecondArticol} />
    <Route exact path={"/thirdarticol"} component={ThirdArticol}
/>
    <Route
        render=(({ location }) => {
            if (location.pathname !== "/") {return <><Header /
><Footer /></>};
        })
    />
</HashRouter>
);
};

```

### **config.js**

```

var config2 = []
const prod = {
    baseUrl: `${window.location.origin}`,
    baseLOCATION: "",
};
const dev = {
    baseUrl: "http://localhost:4000",
    baseLOCATION: "",
};
config2 = process.env.NODE_ENV === `development` ? dev : prod
config2.AppName = 'ECOSYSTEM'
config2.appVersion = '1.00'
export const config = config2
export const appversion = "v1.62";

```

### **index.css**

```

body {
    margin: 0;
    padding: 0;
    font-family: -apple-system, BlinkMacSystemFont, "Segoe UI",
"Roboto", "Oxygen",
    "Ubuntu", "Cantarell", "Fira Sans", "Droid Sans", "Helvetica
Neue",
    sans-serif;
    -webkit-font-smoothing: antialiased;
    -moz-osx-font-smoothing: grayscale;
}
code {

```





```

    font-family: source-code-pro, Menlo, Monaco, Consolas, "Courier
New",
    monospace;
}
.menu {
  display: flex;
  list-style-type: none;
  justify-content: flex-end;
}
.menu li {
  margin: 0 10px;
  cursor: pointer;
  text-decoration: underline;
}

```

### index.tsx

```

import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import './main.scss'
import '../node_modules/bootstrap/dist/css/bootstrap.css'
import registerServiceWorker from './registerServiceWorker';
import { Provider } from 'react-redux'
import { createStore, applyMiddleware } from 'redux'
import reduxThunk from 'redux-thunk'
import reducers from './redux/reducers/index'
import ApolloClient from 'apollo-boost';
import { ApolloProvider } from '@apollo/react-hooks';
import { config } from './config'
import './services/i18n';
import { composeWithDevTools } from 'redux-devtools-extension';
const userName = sessionStorage.getItem('email'),
    role = JSON.parse(sessionStorage.getItem('permisiuni')),
    name = sessionStorage.getItem('name'),
    user = sessionStorage.getItem('userEmail'),
    jwtToken = sessionStorage.getItem('id'),
    nokiaid = sessionStorage.getItem('nokiaid'),
    type = sessionStorage.getItem('type')
const apiclient = new ApolloClient({ uri: config.baseURL +
config.baseLOCATION + `/graphql`, });
console.log({apiclient})
ReactDOM.render( <ApolloProvider client={apiclient}><Provider
  store={createStore(
    reducers,
    {
      auth: {
        name: name ? name : '',
        userName: userName ? userName : '',
        role: role ? role :

```



```

    ',
        user: user,
        token: jwtToken,
        nokiaid: nokiaid,
        type: type,
        isAuthenticated: user ? true : false
    }
  },
  composeWithDevTools(applyMiddleware(reduxThunk))
))
<<App /> </Provider></ApolloProvider>,
document.getElementById('root'));
registerServiceWorker();

```

### main.scss

```

$trans-time: 300ms;
$width: 100%;
$h1-color: #142431;
$bg-color: #424242;
$muted-color: mix(white, $bg-color, 70%);
$button-color: #424242;
body {
  display: flex;
  min-height: 100vh;
  flex-direction: column;
}
.container {
  flex: 1; /* same as flex-grow: 1; */
}
body {
  height: 100%;
  width: 100%;
  margin: 0;
  padding: 0;
  height: 100%;
  width: 100%;
  webkit-scrollbar {display:none;}
}
.upload-container .alert {
  top: 100px;
  text-align: center;
  margin: auto;
  width: 400px;
}
::-webkit-scrollbar {display:none;}
.button {
  background: $button-color;
  color: #fff;
  border: none;
  position: relative;

```



```

    font-size: 1.6em;
    padding: 0 2em;
    cursor: pointer;
    transition: 800ms ease all;
    outline: none;
    margin-bottom: 20px;
}
.button:hover {
    background: #fff;
    color: $button-color;
}
.button:before,
.button:after {
    content: "";
    position: absolute;
    top: 0;
    right: 0;
    height: 2px;
    width: 0;
    background: $button-color;
    transition: 400ms ease all;
}
.button:after {
    right: inherit;
    top: inherit;
    left: 0;
    bottom: 0;
}
.button:hover:before,
.button:hover:after {
    width: 100%;
    transition: 800ms ease all;
}
.dropdown {
    .customButton {
        border: none;
        background: none !important;
    }
    .customButton:focus,
    .customButton:active {
        outline: none !important;
        box-shadow: none;
    }
    .btn-secondary:not(:disabled):not(.disabled):active:focus,
    .show>.btn-secondary.dropdown-toggle:focus {
        box-shadow: none;
    }
    .dropdownCustom {
        margin-right: -20px;
        margin-top: 10px;
        background-color: #414141;
        width: 100%;
    }
}

```



```

        .dropitemCustom:hover {
            background-color: #414141;
        }
    }
}
.wrapper {
    .form-login {
        fieldset {
            width: 400px;
            margin: auto;
        }
    }
    .signing {
        margin: 0 auto;
        width: 80%;
    }
}
.filter {
    margin: 0
}
.legend {
    .box {
        border-top: 3px solid black;
        border-right: 3px solid black;
    }
    .box-bottom {
        border-bottom: 3px solid black;
    }
    .row {
        margin: 0;
        padding: 0;
    }
}
.react-datepicker-wrapper {
    width: 100%;
}
.eventBar {
    margin-left: 0;
    width: 100%;
    background-color: rgba($color: grey, $alpha: 0.05);
}
.request {
    padding: 10% 0;
    margin: auto;
}
.noApprove {
    height: 90vh;
    h1 {
        padding: 20% 0;
        font-size: 12vh;
        color: rgba($color: #000000, $alpha: 0.3);
    }
}

```



```

}
.customSpan {
  min-width: 110px;
}
.modal-footer {
  display: block;
  align-items: center;
  justify-content: unset;
}
.passwordSpan {
  cursor: pointer;
}
@media screen and (max-width: 255px) {
  .container {
    min-width: 100%;
  }
}
.group {
  position: relative;
  margin: 45px 0;
  input {
    background: none;
    color: #343a40;
    font-size: 18px;
    padding: 10px 10px 10px 5px;
    display: block;
    width: $width;
    border: none;
    border-radius: 0;
    border-bottom: 1px solid $muted-color;
    &:focus {
      outline: none;
    }
    &:focus~label,
    &:valid~label {
      top: -14px;
      font-size: 12px;
      color: $hl-color;
    }
    &:focus~.bar:before {
      width: $width;
    }
  }
  input[type="password"] {
    letter-spacing: 0.3em;
  }
  label {
    color: $muted-color;
    font-size: 16px;
    font-weight: normal;
    position: absolute;
    pointer-events: none;

```



```

    left: 5px;
    top: 10px;
    transition: $trans-time ease all;
  }
  .bar {
    position: relative;
    display: block;
    width: $width;
    &:before {
      content: "";
      height: 2px;
      width: 0;
      bottom: 0px;
      position: absolute;
      background: $hl-color;
      transition: $trans-time ease all;
      left: 0%;
    }
  }
}
.checkboxContainer {
  display: inline-flex;
  margin: auto;
}
.checkbox {
  padding-left: 2px;
  padding-right: 2px;
  padding-top: 4px;
  font-size: 1rem;
  font-weight: 400;
  color: #495057;}

```

### **registerServiceWorker.ts**

```

// tslint:disable:no-console
// In production, we register a service worker to serve assets
from local cache.
// This lets the app load faster on subsequent visits in
production, and gives
// it offline capabilities. However, it also means that developers
(and users)
// will only see deployed updates on the 'N+1' visit to a page,
since previously
// cached resources are updated in the background.
// To learn more about the benefits of this model, read https://
goo.gl/KwvDNy.
// This link also includes instructions on opting out of this
behavior.
const isLocalhost = Boolean(
  window.location.hostname === 'localhost' ||
  // [::1] is the IPv6 localhost

```



```

address.
  window.location.hostname === '[::1]' ||
  // 127.0.0.1/8 is considered localhost for IPv4.
  window.location.hostname.match(
    /^127(?:\.(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?) ){3}$/
  )
);
export default function register() {
  if (process.env.NODE_ENV === 'production' && 'serviceWorker' in
navigator) {
    // The URL constructor is available in all browsers that
support SW.
    const publicUrl = new URL(
      process.env.PUBLIC_URL!,
      window.location.toString()
    );
    if (publicUrl.origin !== window.location.origin) {
      // Our service worker won't work if PUBLIC_URL is on a
different origin
      // from what our page is served on. This might happen if a
CDN is used to
      // serve assets; see https://github.com/facebookincubator/
create-react-app/issues/2374
      return;
    }
    window.addEventListener('load', () => {
      const swUrl = `/service-worker.js`;
      if (isLocalhost) {
        // This is running on localhost. Lets check if a service
worker still exists or not.
        checkValidServiceWorker(swUrl);
        // Add some additional logging to localhost, pointing
developers to the
        // service worker/PWA documentation.
        navigator.serviceWorker.ready.then(() => {
          console.log(
            'This web app is being served cache-first by a service
' +
            'worker. To learn more, visit https://goo.gl/SC7cgQ'
          );
        });
      } else {
        // Is not local host. Just register service worker
registerValidSW(swUrl);
      }
    });
  }
}
function registerValidSW(swUrl: string) {
  navigator.serviceWorker
    .register(swUrl)

```



```

.then(registration => {
  registration.onupdatefound = () => {
    const installingWorker = registration.installing;
    if (installingWorker) {
      installingWorker.onstatechange = () => {
        if (installingWorker.state === 'installed') {
          if (navigator.serviceWorker.controller) {
            // At this point, the old content will have been
purged and
            // the fresh content will have been added to the
cache.
            // It's the perfect time to display a 'New content
is
            // available; please refresh.' message in your web
app.
            console.log('New content is available; please
refresh.');
```

```

          } else {
            // At this point, everything has been precached.
            // It's the perfect time to display a
            // 'Content is cached for offline use.' message.
            console.log('Content is cached for offline use.');
```

```

          }
        }
      };
    }
  });
})
.catch(error => {
  console.error('Error during service worker registration:',
error);
});
}

function checkValidServiceWorker(swUrl: string) {
  // Check if the service worker can be found. If it can't reload
the page.
  fetch(swUrl)
    .then(response => {
      // Ensure service worker exists, and that we really are
getting a JS file.
      if (
        response.status === 404 ||
        response.headers.get('content-
type')!.indexOf('javascript') === -1
      ) {
        // No service worker found. Probably a different app.
Reload the page.
        navigator.serviceWorker.ready.then(registration => {
          registration.unregister().then(() => {
            window.location.reload();
          });
        });
      }
    });
}

```





```

    } else {
      // Service worker found. Proceed as normal.
      registerValidSW(swUrl);
    }
  })
  .catch(() => {
    console.log(
      'No internet connection found. App is running in offline
mode.'
    );
  });
});
}
export function unregister() {
  if ('serviceWorker' in navigator) {
    navigator.serviceWorker.ready.then(registration => {
      registration.unregister();
    });
  }
}

```

## Concluzii

Proiectul se poate rezuma la importanța și dorința de a crea un site web adaptabil, acest concept este prezent în majoritatea aplicațiilor web având în vedere importanța unui design ce se pliază pe diverse dispozitive.

Unicitatea acestuia este dată de alăturarea elementelor interactive și construirea lor astfel încât să ofere o experiență plăcută utilizatorului.

Proiectul în starea sa finală coincide cu descrierea inițială, fiind urmați toți pașii prezentați în Tema proiectului. Au fost modificate header-ul, footer-ul și body-ul aplicației, de asemenea au fost adăugate noi funcționalități și stilizate paginile noi.

Una dintre cele mai semnificative provocări în realizarea proiectului a fost învățarea principiilor din React, dar și înțelegerea conceptului responsive.

Pentru a fi un site web complet, mai este necesară definirea unor elemente, iar pentru că mereu se poate mai bine și elementele prezente ar putea fi perfecționate.

**Data prezentării:** 10.01.2022

**Nume:** Savu Roxana Florina

