

Gruppprojektet: Teknisk bidragsrapport Sandra Jakobsson

Mitt bidrag

Planering

Under vårt första uppstartsmöte var vi alla lite osäkra på hur vi ville börja. Med erfarenhet från vår agila kurs föreslog jag att vi skulle börja med att prata om förväntningar och hur vi skulle vilja arbeta.

Vi gick laget runt att alla fick säga hur var och en skulle vilja och kunna jobba ihop. Vi fann snabbt att vi hade samma visioner om vad vi ville skapa och hur vi ville arbeta.

Jag känner själv att jag skulle vilja ha rollen som scrummaster men då jag dels skulle vara bortrest en större del av projektet men framförallt att en av UX:are skulle ha den roll så backade jag därefter i vårt planeringsarbete och lämnade över den rollen.

Fortsatt arbete

Efter vårt uppstartsmöte visade UX:arna upp sin idé i Figma. Vi utvecklare gick igenom och delade upp arbetet mellan oss. För min del (med tanke på att jag skulle vara bortrest en större del av projektet) ville försäkra mig om att jag kunde delta på ett sätt som inte skulle påverka projektet i stort. Jag var nyfiken på och ville själv göra navbar inkl hamburgermeny (detta då jag inte har gjort det innan) och det föll sig därefter naturligt att jag även gjorde footer.

Vidare så skulle jag vilja skapa två funktioner som skulle tillföra projektet men som projektet inte var beroende av:

- En sida med favoriter
- Kunna skapa personliga konton

En annan gruppmedlem skapade en skapa-konton-sida samt logga-in-sida. Dock hann vi ej skapa själva funktionen för konton i vårt backend, vilket jag i efterhand har haft som mål att skapa.

Efter en vecka hade vi kommit mycket långt i vårt arbete, jag personligen hade skapat navbar + footer för mobilversion.

Därefter reste jag iväg. Från början hade jag en ambition att forstätta arbeta under min resa. Men väl på plats så kände jag att jag behövde fokusera på resan. (6 månader innan hade jag blivit uttagen för att representera landslaget på VM i bänkpress i Sydafrika). Med facit i hand så har jag valt att se att detta var ett rätt beslut, dock inte utan konsekvenser. Min grupp var väldigt förstående och har hjälpt mig på alla sätt och vis. Och tack vare dom har vi gjort ett väldigt bra arbete och kommit fram till en mycket bra produkt. Dock känner jag personligen att jag har varit frånvarande och inte har alls lika bra koll som dom andra. Och jag känner att jag inte har bidragit med så mycket som jag själv hade velat och även vad som krävs för projektet.

Efter min resa har jag fortsatt att arbeta med "mina" delar (dessa är dock inte med i presentationen och inte heller i den senaste gemensamma uppdateringen på render).

Jag har fortsatt att skapa en funktion för att favorisera events. Därefter har jag förberett för att kunna skapa konton (via) signup:sidan som Sam har skapat. Jag har i skrivandets stund ett steg kvar och det är att lägga till en ny tabell i vår databas. Klara har ansvarat för vår backend och hon har hjälpt mig vidare i detta arbete men det är inte helt klart än.

Jag har skapat ett eget respiratory på Github och har börjat lägga upp projektet på render men har stött på problem. Jag har försökt lösa problemet några timmar men kommer inte vidare och har därför valt att lämna det för att fokusera på rapporten.

Fortsatt arbete: Jag kommer att jobba vidare med kod (databas) samt koppla ihop personligt konto med mina favoriter. Därefter kommer jag att försöka lösa problemet med Render. I skrivandets stund kommer jag att lämna in det jag har så här långt och därefter komplettera.

Reflektioner

Jag/vi tyckte verkligen om att få en färdig idé/layout. Det gjorde att vi på en gång kunde fokusera på att utveckla. Dock kände jag att jag inte hade tillräckliga kunskaper i Figma för att enkelt kunna läsa av hur jag skulle utveckla. Det tog lång tid för mig att få fram alla detaljer jag behövde. Jag vet fortfarande inte om detta endast berodde på mig, jag skulle kunna tänka mig/känner att det även beror på att UX:arna inte har utvecklat alla komponenter på det sätt jag själv hade behövt.

Till exempel så har dom lagt in bilder/loggor som jag ej har lyckats ta reda på hur jag har kunnat använda. Jag vet inte om det beror på att jag ej kan eller om dom har sparat det i ett annat format.

För att kunna effektivisera arbetet hade jag velat ha mer kunskap i figma och även önskat mer coaching i figma från deras håll.

En annan stor bit i vårt arbete är att jag personligen saknar ett strukturerat arbetssätt. Vi blev tillsagda att arbeta enligt SRUM men att vi själva skulle få välja hur. En av UX-studenterna skulle vara SCRUM-master. Dock fick jag en känsla av att dom inte hade fått tillräckligt med info/utbildning i vad detta innebar. På grund av detta kände jag att vi inte hade ett tydligt strukturerat arbetssätt, vi saknade en plan. Detta är något jag kommer att ta med mig till nästa gruppprojekt.

Trots detta så tycker jag att vi hade ett mycket bra samarbete, vi var lyhörda mot varandra och kunde skapa en enligt mina ögon en mycket bra produkt.

Ps. Jag skulle vilka poängtera och lyfta det faktum att Klara Bryntesson har tagit ett mycket stort ansvar i vårt arbete. Förutom att tekniskt starta vårt projekt har hon även skapat dom viktigaste funktionerna. Förutom det har hon även haft en mycket coachande roll mot mig personligen. Ds.

Länk till projektet

Då jag ej har lyckats publicera mitt eget projekt, därför länkar jag här till gruppens render:
<https://eventer.onrender.com/>

Och därefter till min personliga Github med mina kompletteringar: <https://github.com/Saja81/eventersj>

Min kod

Jag har bidragit med följande kodstycken:

Nav-Bar inklusive hamburgermeny

```
// import { Link } from "react-router-dom"; import HamburgerMenu from "../HamburgerMenu";

function NavBar() { return (

); }

export default NavBar;

Hamburgermeny

import { Link } from "react-router-dom"; import React, { useState } from "react"; import Container from "react-bootstrap/Container"; import Row from "react-bootstrap/Row"; import Col from "react-bootstrap/Col"; import SecondaryButtonNav from "../SecondaryButtonNav";

const HamburgerMenu = () => { const [isOpen, setIsOpen] = useState(false); const [isLogoVisible, setIsLogoVisible] = useState(true); const [isButtonVisible, setIsButtonVisible] = useState(true);

const toggleMenu = () => { setIsOpen(!isOpen); setIsLogoVisible(isOpen); setIsButtonVisible(isOpen); }; return (

{isLogoVisible && ( Eventer )}
```

```
<Col>
  <div className="Nav-bar">
    <nav id="menu" className={isOpen ? "open" : ""}>
      <Row>
        <Col>
          <div className="containerlogo">
            <Link to="/">
              <p id="logo">Eventer</p>
            </Link>
          </div>
        </Col>

        <Col>
          <button
            id="HamburgerButton"
            aria-expanded={isOpen}
            onClick={toggleMenu}
            className={`hamburger1 ${isOpen ? "open" : ""}`} `
          >
            <span className="line"></span>
            <span className="line"></span>
            <span className="line"></span>
            <span className="line"></span>
          </button>
        </Col>
      </Row>
    <Row className="Row-Nav">
```

```

        <ul className="NavBar-links">
          {/* <SearchField /> */}
          <li className="NavBar-link">
            <Link to="">Search</Link>
          </li>

          <li className="NavBar-link">
            <Link to="/eventlist">Evenemang</Link>
          </li>

          <li className="NavBar-link">
            <Link to="/eventlist">Kontakta oss</Link>
          </li>
        </ul>
      </Row>
      <Row className="Row-Nav">
        <ul className="NavBar-links">
          <SecondaryButtonNav>
            <Link to="/login">Logga in</Link>
          </SecondaryButtonNav>

          <li className="NavBar-link">
            <Link to="/signup">Skapa ett konto</Link>
          </li>

          <li className="NavBar-link">
            <Link to="/profile">Mina favoritevent</Link>
          </li>
        </ul>
      </Row>
    </nav>
  </div>
</Col>
<Col className="containerbutton">
  {/* <div> */}
  {isButtonVisible && (
    <button
      id="HamburgerButton"
      aria-expanded={isOpen}
      onClick={toggleMenu}
      className={`hamburger2 ${isOpen ? "open" : ""}`}
    >
      <span className="line"></span>
      <span className="line"></span>
      <span className="line"></span>
      <span className="line"></span>
    </button>
  )}
  {/* </div> */}
</Col>

```

```

    </Row>
  </Container>

```

```
);
```

```
export default HamburgerMenu;
```

Footer

```
import { Link } from "react-router-dom"; import Container from "react-bootstrap/Container"; import Row from "react-bootstrap/Row"; import Col from "react-bootstrap/Col";
```

```
import { Instagram, Twitter, Facebook, Youtube, Tiktok, } from "react-bootstrap-icons";
```

```
function Footer() { return (
```

```
  Eventer
```

```
  Följ oss
```

```

    <Col>
      <Youtube />
    </Col>
    <Col>
      <Tiktok />
    </Col>
  </Row>
  <Row>
    <div className="footer">
      <ul className="Footer-list">
        <li>
          <Link className="Footer-links" to="/login">
            Logga in
          </Link>
        </li>
        <li>
          <Link className="Footer-links" to="/signup">
            Skapa konto
          </Link>
        </li>
        <li>
          <Link className="Footer-links" to="/">
            Mina sidor
          </Link>
        </li>
        <li>
          <Link className="Footer-links" to="/EventWiew">
            Evenemang
          </Link>
        </li>
      </ul>
    </div>
  </Row>

```

```

        <li>
          <Link className="Footer-links" to="/">
            Kontakt
          </Link>
        </li>
        <li>
          <Link className="Footer-links" to="/">
            FAQ
          </Link>
        </li>
      </ul>
    </div>
  </Row>
  <Row>
    <p>Våra samarbetspartners</p>
  </Row>
  <Row>
    <Col>
      </img>
    </Col>
    <Col>
      </img>
    </Col>
    <Col>
      </img>
    </Col>
  </Row>
</Container>
</div>

```

```
); }
```

```
export default Footer;
```

Profileview

```
import React, { useState, useEffect } from "react";

function ProfileView({ favouriteEventName }: { favouriteEventName: string }) { const [favouriteEvents,
setFavouriteEvents] = useState<string[]>([]);

useEffect(() => { const savedEvents = localStorage.getItem("favouriteEvents"); if (savedEvents) {
setFavouriteEvents(JSON.parse(savedEvents)); } }, []);

useEffect(() => { if (favouriteEventName && !favouriteEvents.includes(favouriteEventName)) {
setFavouriteEvents((prevFavouriteEvents) => { const updatedEvents = [...prevFavouriteEvents,
favouriteEventName]; localStorage.setItem("favouriteEvents", JSON.stringify(updatedEvents)); return
updatedEvents; }); } }, [favouriteEventName, favouriteEvents]);

function removeAllFavourites() { localStorage.removeItem("favouriteEvents"); setFavouriteEvents([]); }

return (
```

Hej Sandra

```
<h4>Det här är dina favoritevent:</h4>
<div>
  <ul>
    {favouriteEvents.map((event, index) => (
      <li key={index}>{event}</li>
    ))}
  </ul>
</div>

  <button onClick={removeAllFavourites}>To bort all favoritevent </button>
</div>
```

```
); }
```

```
export default ProfileView;
```

Spara kononton

OBS: ENDA POSTANROPET ÄR TILLAGT AV MIG

```
import express, { Application, Request, Response, NextFunction } from "express"; import cors from "cors";
import dotenv from "dotenv"; import path from "path"; import { Client } from "pg"; import mime from "mime-
types";
```

```
dotenv.config();
```

```
const client = new Client({ database: process.env.PGDATABASE, host: process.env.PGHOST, password:
process.env.PGPASSWORD, port: process.env.PGPORT ? parseInt(process.env.PGPORT) : undefined, user:
process.env.PGUSER, });
```

```
client.connect();
```

```
const app: Application = express();

const port = process.env.PORT || 8080; app.use(cors());

app.use(express.json());

app.use(express.urlencoded({ extended: false }));

app.use(express.static(path.join(__dirname, "public"))); app.use("/images", express.static(path.join(__dirname, "publicimages")));

app.use((req, res, next) => { const filePath = path.join(__dirname, "public", req.url); const contentType = mime.contentType(path.extname(filePath)) || "application/octet-stream"; res.type(contentType); next(); });

app.get("/static/js/:filename", (req, res, next) => { const filePath = path.join( __dirname, "public", "static", "js", req.params.filename ); res.set("Content-Type", "application/javascript"); res.sendFile(filePath); });

app.get( "/events", async (request: Request, response: Response, next: NextFunction) => { const { rows } = await client.query("SELECT * FROM events"); if (rows.length === 0) { response.status(404).end(); } else { response.status(200).send(rows); } });

app.get( "/events/:eventId", async (request: Request, response: Response, next: NextFunction) => { const { rows } = await client.query( "SELECT * FROM events WHERE id_name = $1", [request.params.eventId] );
```

```
    if (rows.length === 0) {
        response.status(404).end();
    } else {
        const event = rows[0];
        response.status(200).send(event);
    }
});
```

```
app.get( "/openhours/:eventId", async (request: Request, response: Response, next: NextFunction) => { const { rows } = await client.query( "SELECT * FROM openhours WHERE event_id = $1", [request.params.eventId] );
```

```
    let event = null;
    if (rows.length === 0) {
        event = null;
        response.status(404).end();
    } else {
        event = rows[0];
        response.status(200).send(event);
    }
});
```

```
}); // postanrop tillagts av Sandra app.post("/signup", async (request: Request, response: Response, next:NextFunction) => { const { name, email, password } = request.body;
```



```
try { const { rows } = await client.query( "INSERT INTO users (name, email, password) VALUES ($1, $2, $3)
RETURNING *", [name, email, password] );
```

```
    const newUser = rows[0];
    response.status(201).send(newUser);
```

```
  } catch (error) { response.status(500).send(error); } });
```

```
app.listen(port, () => { console.log("Redo på " + port); });
```

```
// app.listen(8080, () => { // console.log("Redo på http://localhost:8080/"); // });
```

CSS

```
@font-face { font-family: Fontname; src: url("Fonts/holligate-signature-demo/holligate-signature-demo.ttf")
format("truetype"); font-weight: normal; font-style: normal; }
```

```
.App { text-align: center; }
```

```
.gradient { background: linear-gradient( to right, rgba(239, 118, 87, 1), rgba(255, 168, 123, 0.8134), rgba(251,
46, 95, 0.41) ); }
```

```
.Nav-bar { display: flex; justify-content: center; }
```

```
#menu { display: none; width: 358px !important; height: 374px !important; }
```

```
.NavBar-links { color: #ffffff; text-decoration: none; font-family: Arial, Helvetica, sans-serif; text-align: center; }
```

```
.NavBar-link { padding-top: 20px; }
```

```
.footer { margin-top: 20px; }
```

```
.Footer-links { color: black; text-decoration: none; font-family: Arial, Helvetica, sans-serif; padding-left: 16px;
padding-top: 15px; }
```

```
.Footer-list { margin-left: -50px; }
```

```
.App-header { min-height: 100vh; display: flex; flex-direction: column; align-items: center; justify-content:
center; font-size: calc(10px + 2vmin); }
```

```
.hamburger1, .hamburger2 { width: 40px; height: 40px; border: none; background-color: transparent; cursor:
pointer; display: flex; flex-direction: column; justify-content: center; align-items: center; } #HamburgerButton {
margin-left: 100px; }
```

```
#HamburgerButton.open { margin-left: 100px; }
```

```
.line { width: 30px; height: 2px; background-color: white; margin: 4px 0; transition: transform 0.3s ease-in-out; }
```

```
.open .line:nth-child(1) { opacity: 0; }
```

```
.open .line:nth-child(2) { transform: rotate(45deg) translate(45%, -45%); } .open .line:nth-child(3) { opacity: 0; }
```

```
.open .line:nth-child(4) { transform: rotate(-45deg) translate(45%, -45%); }

.containerlogo { display: flex; justify-content: flex-start; }

#headerrow { padding-top: 20px; }

#logo, #logo1 { font-family: "Fontname"; color: white !important; font-size: 36px !important; }

#logo1:hover { color: #eeeeee; }

#logo:hover { color: #eeeeee; }

#menu.open { display: block; background-color: #e06943; border: solid; border-color: #e06943; border-radius: 4%; width: 80%; margin-bottom: 10px; }

li { list-style: none; padding-left: -50px; }
```