

Projekt 1 - Todo liste

2. Semester

Marc Kastholm

Amir Skoric

Lasse Kjellerup

Anders Adriansen

17.2.2023

GITHUB LINK: <https://github.com/andersravn32/todo-list>

Indhold

Indledning:	3
Problemformulering:	3
Metodeovervejelser:	4
Research:	5
Frontend	5
Backend	6
Analyse:	7
Konstruktion:	8
Evaluering af process:	9
Konklusion:	10
Referencer:	11

Indledning:

Datasikkerhed er et term der har været til stede siden starten af internettet. Et term der er med til at sikre brugernes anvendelse af internettet og give dem og deres personlige oplysninger en sikkerhed som forhindrer hackere fra at indhente oplysninger og anvende dem til uautoriserede formål. Det er derfor af afgørende betydning at have effektive sikkerhedsforanstaltninger på plads for at beskytte data mod uautoriseret adgang, manipulation og ødelæggelse.

Vi har med denne viden om datasikkerhed udarbejdet en applikation med en dokumentbaseret database, der er designet til at være fleksibel og skalerbar, og som giver mulighed for hurtig og effektiv lagring og hentning af data. Alt dette er opnået ved anvendelse af Vue.js, MongoDB, Express.js og Node.js.

Med implementeringen af effektive sikkerhedsforanstaltninger, såsom salting og hashing af bruger passwords, har vi taget databeskyttelse seriøst og prioriteret vores brugeres personlige oplysninger. Vi er stolte af at have udviklet en applikation, hvor brugeroplevelsen går hånd i hånd med datasikkerhed. Vi vil yderligere undersøge i vores udarbejdede problemformulering for at give jer et dybere indblik i vores arbejde med at udvikle en sikker og brugervenlig "to do" applikation.

Problemformulering:

Hvordan kan vi udarbejde en "todo" liste app, med en dynamisk brugergrænseflade med dertilhørende backend?

Metodeovervejelser:

Vi benyttede den agile arbejdsmetode under hele processen, hvilket tillod os at tilpasse os ændringer og justere vores tilgang. Dette var afgørende i udviklingen af vores applikation, hvor vi løbende evaluerede og justerede vores teknologivalg og design for at imødekomme kravene fra projektbeskrivelsen.

Valget af MongoDB som vores database var baseret på dens fleksible datamodel, som muliggjorde en mere løs struktur end traditionelle relationelle databaser som MySQL. Dette gjorde det hurtigere og nemmere at implementere ændringer og justere designet af vores applikation.

Vue.js blev anvendt til udvikling af vores frontend applikation, hvilket gav os mulighed for at opbygge en dynamisk brugergrænseflade væsentligt nemmere og hurtigere end med "vanilla" JavaScript. Vi brugte også TailwindCSS til styling af vores brugergrænseflade, hvilket muliggjorde mere effektiv udvikling ved at udnytte komponentstrukturen i Vue.js.

Node.js og Express.js blev valgt som primære værktøjer til vores backend, da de gav os mulighed for at udvikle en fleksibel og skalerbar løsning og arbejde med open source teknologier til håndtering af sessions og tokens. Vi valgte også at implementere bcrypt-metoden til at hashe og salte brugerpasord for at sikre, at brugernes personlige oplysninger var ordentligt beskyttet.

Alt i alt var vores valg af værktøjer og metoder afgørende for udviklingen af en sikker og brugervenlig "to do" applikation, der opfyldte kravene fra projektbeskrivelsen og leverede en positiv brugeroplevelse.

Research:

Før vi kunne påbegynde udviklingsfasen af dette projekt skulle vi beslutte os for hvilke teknologier vi skulle benytte til projektet. Dette har vi besluttet i to forskellige omgang, fordelt ud på frontend- og backend udvikling, og hvilke teknologier som ville passe de forskellige dele.

Frontend

Til udviklingen af frontend applikationen har vi benyttet os af VueJS som JavaScript framework, da det har muliggjort at vi har kunne udvikle en dynamisk brugergrænseflade væsentligt nemmere og hurtigere end vi ville have været i stand til såfremt vi skulle udvikle den ved brug af "vanilla" JavaScript. Udover VueJS til udvikling har vi også benyttet os af et CSS framework, ved navn TailwindCSS, til styling af vores brugergrænseflade. Dette har medvirket til at vi har kunne udnytte komponent strukturen i VueJS til sit fulde potentiale, da VueJS og TailwindCSS spiller rigtig godt sammen.

I udviklingen af frontend applikationen valgte vi Vue.js, da det er et populært JavaScript framework, der gør det muligt at bygge brugergrænseflader på en nem og effektiv måde. Vue.js har et stort udviklermiljø og en fleksibel og skalerbar struktur, hvilket gjorde det til det ideelle valg for vores projekt. Vue.js er også kendt for at have en god ydeevne, da det er designet til at minimere antallet af genindlæsninger på hjemmesiden og gøre den mere responsiv for brugeren.

Vi valgte også at benytte os af TailwindCSS til styling af vores brugergrænseflade, da det er et CSS framework, der gør det nemt at designe og style en hjemmeside på en effektiv måde. TailwindCSS er kendt for at have en intuitiv og fleksibel struktur, der gør det nemt at tilpasse designet efter vores behov. Ved at benytte

os af TailwindCSS har vi været i stand til at reducere mængden af CSS-kode, som vi skulle skrive, og samtidig opnå en mere effektiv udvikling af frontend applikationen.

Backend

Til udvikling af backend applikationen har vi valgt at benytte Node.js som vores primære runtime og Express.js som vores HTTP framework til at modtage og håndtere anmodninger fra vores frontend. Vi valgte denne kombination, da det gav os mulighed for at udvikle en fleksibel og skalerbar løsning. Node.js er kendt for sin evne til at håndtere store mængder af samtidige anmodninger effektivt, mens Express.js gav os et simpelt, men kraftfuldt værktøj til at opbygge vores RESTful API.

Som vores database valgte vi MongoDB på grund af dens fleksible datamodel, der gjorde det muligt at arbejde med en mere løs struktur end traditionelle relationelle databaser som MySQL. Dette valg gav os større frihed til at implementere ændringer og justere designet af vores applikation, samtidig med at det hjalp os med at undgå de kompleksiteter, som ofte er forbundet med relationelle databaser.

Vores backend-løsning giver brugerne mulighed for at oprette, redigere og slette to-do-punkter. Vi har implementeret bcrypt-metoden til at hashe og salte brugerpasswords for at sikre, at brugernes personlige oplysninger er ordentligt beskyttet. Vi har også benyttet forskellige open source teknologier til håndtering af sessions og tokens, der har hjulpet os med at opretholde en sikker og pålidelig brugeroplevelse.

Alt i alt har vores valg af teknologier og metoder i backend-udviklingen været afgørende for udviklingen af en sikker, pålidelig og skalerbar løsning. Ved at tage hensyn til faktorer som fleksibilitet, effektivitet og databeskyttelse har vi opnået

en løsning, der opfylder projektbeskrivelsens krav og samtidig leverer en positiv brugeroplevelse.

Analyse:

Vores projekt er en "to do" applikation, der er udviklet til at hjælpe brugere med at organisere deres daglige aktiviteter og opgaver. Gennem vores applikation ønsker vi at tilbyde en intuitiv og brugervenlig oplevelse for brugerne, så de kan føle sig mere produktive og organiserede i deres daglige liv.

For at opnå dette mål har vi valgt at udvikle applikationen med moderne teknologier og værktøjer. I frontend udviklingen har vi benyttet VueJS, som har gjort det nemmere og hurtigere at udvikle en dynamisk brugergrænseflade end hvis vi havde brugt "vanilla" JavaScript. Derudover har vi anvendt TailwindCSS, som er et effektivt CSS framework, der har tilladt os at opnå en effektiv og struktureret udvikling af frontend.

I backend udviklingen har vi valgt Node.js som vores primære runtime og Express.js som vores HTTP framework. Disse valg blev truffet på grund af deres hastighed og effektivitet, og gjorde det muligt for os at modtage og håndtere anmodninger fra vores frontend. For at gemme og organisere data i vores applikation har vi valgt MongoDB som vores database. Dette valg tillod os at have en mere løs databasestruktur end traditionelle SQL databaser, og er let at skalere, hvilket er vigtigt for at give vores applikation mulighed for at håndtere en stigende mængde brugere.

Vi har også implementeret flere sikkerhedsforanstaltninger for at beskytte vores brugeres personlige oplysninger. Vi anvendte bcrypt-metoden til at hashe og salte brugerpasord, før de blev gemt i databasen. Dette gjorde det vanskeligt for uautoriserede personer at få adgang til brugeres konti, selv hvis de fik adgang til databasen. Vi har også anvendt forskellige open source teknologier til håndtering af sessions og tokens, hvilket yderligere har styrket vores applikations sikkerhed.

Alt i alt har vores valg af teknologier og sikkerhedsforanstaltninger gjort det muligt for os at udvikle en effektiv og sikker "to do" applikation. Vi mener, at vores applikation er en god løsning for brugere, der ønsker at holde styr på deres daglige opgaver og aktiviteter.

Konstruktion:

I dette projekt har vi udviklet en "to do" applikation, hvor brugere kan planlægge og gemme deres aktiviteter i en database. Vi har valgt at bruge MongoDB som vores database og har udviklet applikationen ved hjælp af Node.js og Express.js. For at sikre brugernes personlige oplysninger, har vi implementeret en sikker login-funktion, hvor kun brugere med et unikt bruger-id og et sikkert password kan logge ind og bruge applikationen. For at forbedre brugeroplevelsen har vi udviklet en intuitiv brugerflade, som gør det nemt for brugere at oprette, vedligeholde og se deres "to dos". Applikationen er udviklet med henblik på at være et nyttigt personligt redskab for brugerne.

For at sikre brugernes adgangskoder, har vi anvendt bcrypt-metoden til at hashe og salte passwordene, før de gemmes i databasen. Dette gør det vanskeligt for uautoriserede personer at få adgang til brugernes konti, selvom de skulle få adgang til databasen. Bcrypt er en stærk hash-algoritme, der tager tid at bryde,

og dens styrke kan justeres ved at øge antallet af iterationer, der anvendes under hashingen. Ved at inkludere bcrypt i vores applikation, har vi gjort vores brugeres sikkerhed til en prioritet, og sørget for at deres adgangskoder er ordentligt beskyttet.

Evaluering af process:

Generelt set var processen med at udvikle to-do-applikationen en positiv oplevelse. Teamet arbejdede godt sammen og samarbejdede effektivt for at nå de fastsatte mål. Der var en god kommunikation på tværs af teamet, og alle var engagerede i at levere en kvalitetsapplikation.

En af de positive aspekter ved processen var valget af værktøjer og teknologier. Ved at bruge VueJS og TailwindCSS til frontend udviklingen, kunne teamet skabe en brugergrænseflade, der var både funktionel og æstetisk tilfredsstillende. Node.js og Express.js viste sig også at være gode valg til backend udviklingen, da de gjorde det muligt at opbygge en sikker og pålidelig infrastruktur til håndtering af brugersessioner og adgangskoder.

I denne udviklingsproces har vi formået at undgå betydelige udfordringer, hvilket har været en fordel for projektet. Vi var i stand til at arbejde effektivt og nå vores mål, takket være vores grundige planlægning, tætte samarbejde og forståelse for de teknologier og værktøjer, vi har anvendt. Der var dog nogle mindre udfordringer undervejs, som hurtigt blev løst takket være vores teamindsats og tekniske færdigheder.

Alt i alt var processen med at udvikle to-do-applikationen en succes, da teamet var i stand til at levere en funktionel og brugervenlig applikation inden for de fastsatte tidsfrister. Der var dog nogle udfordringer undervejs, som teamet vil tage med sig i fremtidige projekter.

Konklusion:

I denne opgave har vi udviklet en "to do" applikation ved hjælp af en række forskellige teknologier. Vi har brugt VueJS og TailwindCSS til at udvikle en dynamisk og intuitiv brugergrænseflade, som gør det nemt for brugere at oprette, vedligeholde og se deres "to dos". På backend-siden har vi brugt Node.js og Express.js til at modtage og håndtere anmodninger fra frontend og MongoDB som vores database. Vi har også anvendt bcrypt-metoden til at hashe og salte brugernes adgangskoder, før de gemmes i databasen, for at beskytte deres personlige oplysninger.

I løbet af udviklingen af denne applikation har vi lært, hvordan man kan kombinere forskellige teknologier for at skabe en robust og brugervenlig applikation. Vi har også lært vigtigheden af sikkerhed og beskyttelse af brugernes personlige oplysninger. Ved at inkludere bcrypt i vores applikation har vi sikret, at brugernes adgangskoder er beskyttet, selvom databasen skulle blive kompromitteret.

Samlet set er vi tilfredse med resultatet af vores arbejde. Vi mener, at vi har skabt en effektiv og brugervenlig "to do" applikation, som kan være til gavn for brugere, der ønsker at organisere deres dagligdag på en enkel og effektiv måde. Vi har også fået erfaring med en række forskellige teknologier og udviklingsmetoder, som kan være nyttige i fremtidige projekter.

Referencer:

Web Technology for Developers | MDN (2023) available from
<<https://developer.mozilla.org/en-US/docs/Web>> [16 February 2023]

Introduction | *Vue.js* (n.d.) available from
<<https://vuejs.org/guide/introduction.html>> [16 February 2023]

Installation - Tailwind CSS (n.d.) available from
<<https://tailwindcss.com/docs/installation>> [16 February 2023]

Node.js (n.d.) *Documentation* [online] available from
<<https://nodejs.org/en/docs/>> [16 February 2023]

Express - Node.js Web Application Framework (n.d.) available from
<<https://expressjs.com/>> [16 February 2023]

MongoDB Documentation (n.d.) available from
<<https://www.mongodb.com/docs/>> [16 February 2023]

Bcrypt (2022) available from <<https://www.npmjs.com/package/bcrypt>>
[16 February 2023]

auth0.com (n.d.) *JWT.IO* [online] available from <<http://jwt.io/>>
[16 February 2023]