­­­­

Prosjektrapport

Årsoppgave

August Brynsrud

2023

**Innhold**

[Innledning 2](#_Toc136028030)

[Prosjektbeskrivelse 2](#_Toc136028031)

[Prosjektplan 3](#_Toc136028032)

[Viktige valg 5](#_Toc136028033)

[Java 5](#_Toc136028034)

[Apache/php 5](#_Toc136028035)

[Postgresql 5](#_Toc136028036)

[Debian 6](#_Toc136028037)

[Brukerstøtte tiltak 7](#_Toc136028038)

[Opplæring 7](#_Toc136028039)

[Universell utforming 7](#_Toc136028040)

[Dokumentasjon 9](#_Toc136028041)

[Nettverksdiagram 9](#_Toc136028042)

[Fildiagram 9](#_Toc136028043)

[Bruker Informasjon 10](#_Toc136028044)

[Kartlegging av lovverk 11](#_Toc136028045)

[Risikoanalyse og tiltaksplan 13](#_Toc136028046)

[Forebygging av datatap 15](#_Toc136028047)

[Egenevaluering 17](#_Toc136028048)

[Kilder 19](#_Toc136028049)

# Innledning

## Prosjektbeskrivelse

I årsoppgaven min har jeg laget et spill i kodespråket Java. Spillet er et tekstbasert rollespill med visuelle elementer som får det til å virke som du går rundt i en labyrint. Rommene og etasjene i spillet er laget slik at det er tilfeldig hvordan de ser ut og hva som er inne i dem. Spillet har lokale filer med alle våpen, rustninger, velsignelser og fiender. Når man er på startmenyen, kan man velge å oppdatere de lokale filene og overskrive med informasjon fra en database. Spillet startet enkelt med kun en loop som konstant lagde nye senarioer og lagde nye utfordringer for spilleren, og forkastet gamle etter som spilleren gikk fremover, men senere utviklet jeg et kart med koordinatsystem og mulighet til å utforske en hel labyrint, med uendelig mange etasjer. Nå har brukeren en inventory (en mulighet til å frakte med seg utstyr i spillet) og han eller hun kan selge og kjøpe ting med en in game currency (spillvaluta).

Jeg har også en nettside. Der kan sluttbrukere stille egne spørsmål og finne svar på ting de lurer på i FAQ delen. De kan også se en kort introduksjonsvideo som viser hvordan spillet fungerer. Om man er en admin og jobber på prosjektet, kan man velge å logge inn øverst til venstre. Da får man tilgang til ulike verktøy avhengig av hvilken del av prosjektet man jobber med. For eksempel kan de som jobber med spillet endre gjenstander, og derer attributter gjennom nettsiden. De som jobber med selve nettsiden eller med brukerstøtte kan svare og administrere FAQ spørsmål og svar. Dette gjorde jeg som et sikkerhetstiltak slik at det var minst mulig direkte kontakt med databasen (i utgangspunktet skal alle endringer gjøres gjennom nettsiden).

Til slutt har jeg også en utvikler manual for at noen andre kan enkelt overta prosjektet med en god forståelse om hva det innebærer og hva mine tanker om videre utvikling er.

Hele prosjektet ligger på GitHub: <https://github.com/AugustSabr/Arsoppgave>

# Prosjektplan

Jeg syntes det var vanskelig å legge et tidsaspekt på når jeg gjør de forskjellige delene av oppgaven. Noen dager er jeg litt ineffektiv, andre dager jobber jeg mye både på skolen og hjemme. Derfor valgte jeg å heller skrive en slags check-liste hvor jeg skriver ned hva jeg har tenkt til å gjøre. Jeg har delt oppgavene i fire bolker; prosjektrapport og teknisk dokumentasjon, server-pc, nettside, og selve spillet. Mange av punktene er avhenge av punkter i en annen bolk så jeg hopper litt mellom dem når jeg jobber.

**Prosjektrapport og teknisk dokumentasjon**

* viktige valg og prioriteringer
  + hvilke kodespråk?
  + hvilken server?
* hvilken database?
* risikoanalyse
  + tabell
  + back-up rutiner
* lovverk
  + liste opp relevante lovverk
  + forklare hvordan de påvirker prosjektet mit
* nettverksdiagram
  + et visuelt diagram som viser hvordan nettverket rundt prosjektet mitt er satt opp
* annen teknisk dokumentasjon
  + Arvids exel ark
* egenevaluering
* kilder

**Server pc**

* debian
* apache
  + github
* postgres
* reverse proxy?
* automatisk backup?

**Nettside**

* faq
  + sende inn og svare på spørsmål?
* innlogging
  + autentisering
  + autorisering
* opplæringsmatriell
  + brukerveiledning til utvikler
  + intoduksjonsvideo til sluttbruker

**Spill**

* visuelt med bilder
  + lage alle bilder
* max health potion
* blessings
* expiriance and stat system
* loot enemies?
* luck and crit

# Infrastruktur og viktige valg

Oppgaven min består av selve spillet på sluttbrukers maskin, og apache-nettsiden og postgres-databasen på debian serverpc. Spillet er klientsiden, den skal være enkel å bruke og skal kunne brukes både online og offline. Apacheserveren kan både brukes av sluttbrukere som stiller spørsmål eller ser en video om spillet, og utviklere som svarer på spørsmål eller oppdaterer spillobjektene gjennom databasen på spillet. Postgres-databasen brukes både av spillet og av nettsiden: sluttbrukeren henter informasjon derfra, mens utvikleren oppdaterer informasjonen.

## Java

Spillet er nesten bare skrevet i java med litt SQL for å hente info fra databasen. Java er et objekt-orientert programmeringsspråk som brukes mye til backend development. Java har ganske streng syntaks, så selv med bare små feil nekter programmet å kjøre. Fordelen med Java er at man tvinges til å løse småfeil etterhvert som de dukker opp. Da minsker faren for at det oppstår store feil senere, som krever mye arbeid for å lokalisere og løse. Det er også enklere å lære syntaks og hvordan språket funker når jeg har faste regler å følge i motsetning til python og javascript hvor ting som å ende setninger med «;» er valgfritt.

## Apache/php

Nettsiden er hovedsaklig skrevet i php, med litt html og css. Php brukes mye til å lage dynamiske nettsider. I php kan man både programmere mot databaser, men også skrive ren html kode. Jeg har valgt å nesten kun style sluttbrukers del av nettsiden for å fokusere på at den skal være enkel å bruke for alle.

## Postgresql

Jeg valgte PostgreSQL på grunn av sin skalerbarhet, pålitelighet, og åpen kildekode. Den kan også håndtere store datamengder og komplekse spørringer effektivt. Den er også plattformuavhengig så den kan brukes av nesten alle operativsystemer og har et aktivt samfunn som kan svare på spørsmål. Databasesystemet støtter også ulike datatyper, det var egt det som var hovedgrunnen til byttet mitt. Jeg har jo visuelle elementer på spillet mitt så målet var å legge inn bilder på databasen.

## Debian

Debian er laget med åpen kildekode, og er Linux-basert og mye brukt på servere og database. Jeg hadde lite erfaring med Linux før dette og det var en god mulighet til å lære oprativsystemet. I motsettning til andre systemer som ubuntu som er basert på debian, kommer ikke debian med oppdateringer for de er nøye testet, så du kan være sikker på at den ikke har noen fundamentale feil. Samlet sett er Debian en pålitelig og tilpasningsdyktig løsning for hosting av Apache-server og PostgreSQL-database

# Brukerstøtte tiltak

## Opplæring

Jeg har en manual for utvikler som kort oppsummerer hvordan man kan sette opp prosjektet fra bunnen av. Det burde være ganske enkelt fordi jeg tar ofte backup som ligger på github og jeg har et bashscript med alle kommandoene som er nødvendig for å sette opp en ny debian maskin med apahe og PostgreSQL.For sluttbruker skal jeg lage en video som kort viser litt av spillet for å tiltrekke seg nye brukere. Her finner du opplæringsmateriell: <https://github.com/AugustSabr/Arsoppgave/tree/main/Oppl%C3%A6ringsmatriell>

## Universell utforming

Jeg har også brukt universell utforming og design til å gjøre både spillet og nettsiden intuitiv og enkel å bruke. For eksempel har jeg bare brukt svart og hvit som er en stor kontrast sånn at også fargeblinde kan spille spillet, eller drifte nettsiden. Opplæringsmaterialene for sluttbruker og utvikler, samt kommentarer i koden, gir en god forståelse av prosjektet.

**Dette er designet til spillet:**



Dette er hvordan det ser ut nå:

A screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a video game

Description automatically generated with medium confidence

Spillet er utviklet så det kan spilles uten mus/med kun tastetrykk, da kan det senere enkelt flyttes over til konsoller som ps4, nintendo switch, og xbox for å nå flest mulig spillere.

# Et bilde som inneholder tekst, skjermbilde, diagram, Parallell Automatisk generert beskrivelse Dokumentasjon

## Nettverksdiagram

Her er et eksempel på hvordan brukere kan oppdatere spillet/gå inn på nettsiden fra deres lokale nettverk over internettet og hjem til meg hor server-pc-en ligger.  
Jeg har brukt noe som heter portforwarding som gjør at hvis noen prøver å koble seg på ruteren hjemme gjennom den offentlige IPv4 adressen med en av de godkjente portene blir de videre sendt til server-pc-en min.

Jeg har også brukt noe som heter DMZ som offentligjør kun en liten del av et nettverk, og lar det som ligger der ta hånd om all kommunikasjon mellom mitt lokale nettverk og alt utenfor

## Fildiagram

Diagram

Description automatically generatedDette er et fildiagram som viser hvordan filene i spillet kaller hverandre. Dette brukte jeg for å se om det var noen unødvendige kallinger som betyr at det er ineffektiv kode.

## Bruker Informasjon



# Kartlegging av lovverk

Jeg har kommet fram til syv relevante lovverk som kan være viktige nå eller senere dersom prosjektet vokser.

**Arbeidsmiljøloven:**

* Arbeidsmiljøloven passer på rettighetene til arbeidstaker. Det angår ting som arbeidstider og pauser, arbeidsmiljø og sikkerhet, og stillingsvern. Arbeidsmiljøloven trumfer det som står i arbeidskontrakten så man er beskyttet av loven uansett.
  + Dette er ikke helt relevant til mitt prosjekt nå, men ville ha vært viktig dersom jeg hadde jobbet videre på det og hadde begynt å ansette folk.

**Universell utforming:**

* Universell utforming går ut på at løsninger kan brukes av så mange mennesker som mulig på en likeverdig måte. Den er der for å motvirke diskriminering på grunnlag av nedsatt funksjonsevne
  + I mitt prosjekt er all tekst svart på hvitt eller hvitt på svart så det er tydelige kontraster som tar hensyn til svaksynte og fargeblinde

**Personopplysningsloven (GDPR):**

* Personopplysningsloven handler om behandling, innsamling og bruk av personopplysninger
  + Jeg samler kun informasjon to steder i prosjektet mitt: når sluttbruker sender inn spørsmål, og når en admin logger inn lagres tidspunkt og tidssone. Når sluttbruker sender spørsmål står det tydelig at informasjon kan offentligjøres, så de må være bevisst på hva de skriver. Det må uansett godkjennes av en admin før vi bruker informasjonen til noe.

**Ekomloven (lov om elektronisk kommunikasjon):**

* Ekomloven inneholder blant annet informasjonskapsler (HTTP cookies), som brukes for at en nettside kan gjenkjenne deg (browseren din) senere
  + Når en admin logger in på siden min, opprettes en informasjonskapsel kalt PHPSESSID som brukes for å se privilegiene til adminbrukeren. Privilegiene hentes fra en database og er ikke noe personlig informasjon, uansett slettes denne informasjonskapselen når du logger ut eller lukker vinduet

**Markedsføringsloven:**

* Markedsføringsloven beskytter blant annet barn mot målrettet reklame. Den sier også at man ikke kan reklamere for rusmidler som røyk og alkohol, eller våpen
  + Om nettsiden min skal ha reklame på seg senere må jeg ha i bakhodet at spillet kan spilles av personer på alle aldre og samme kan nettsiden. Jeg må være ekstra bevisst på hva jeg har på siden fordi barn som liker spillet vil bruke den.

**Åndsverkloven (lov om opphavsrett og åndsverk):**

* Åndsverklovenpasser på rettighetene til en skaper. Alt originalt som lages faller automatisk under opphavsrett til et åndsverk i det en skaper verket. Ingen kan bruke andres verk uten en bindende avtale.
  + Alle verk, alle bilder og kode, i mitt prosjekt er originale og er laget av meg, jeg har kun søkt opp hvordan jeg skal løse syntaksproblemer, og når jeg lurte på hvilke parameter visse funksjoner trenger. For eksempel trenger pg\_connet() ip, port, databasenavn, brukernavn, og brukerpassord for å starte en kobling til databasen.

**Straffeloven:**

* Straffeloven er ganske stor men inkluderer blant annet diskriminering og hatefulle ytringer.
  + Når jeg evt. publiserer nettsiden min må jeg være ekstra bevist på hva som står på den. Spørsmål fra brukere må selvsagt godkjennes før de publiseres på siden for å unngå hatefulle ytringer eller at noen deler sensitiv informasjon.

# Risikoanalyse og tiltaksplan

Her er et skjema og så er det litt refleksjon lenger nede.

**Linjeleder: August Brynsrud Dato: 26.05.2023**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Mulig uønsket hendelse/  belastning | Vurdering av sannsynlighet | Vurdering av konsekvens:  (A-E) | | | | Risiko-verdi | Kommmentarer/status  Forslag til tiltak |
| ID nr | (1-5) | Menneske | Ytre miljø | Øk/  materiell | Om-dømme |
| 1 | Informasjon på databasen blir stjålet av en hacker | 2 | A | C | C | C | C | Databasen inneholder ikke mye sensitiv informasjon, kun spillobjekter og brukernavn, hvor passord er hashet, men vi bør uansett stenge ned databasen midlertidig og endre innloggingsinformasjon til databasen, og vurdere og endre innloggingsinformasjonen til nettsiden selv om det er kryptert. |
| 2 | Spillet krasjer under bruk | 4 | A | A | B | B | A | Spillet testes nøye hver gang det kommer en ny endring, så det vil trolig ikke skje. Men om det skjer, må vi komme med en ny versjon. Kanskje også legge ut en melding på nettsiden som viser at vi jobber med problemet |
| 3 | Feilinformasjon gis til spilleren/sluttbruker | 3 | A | B | A | C | B | Om vi får varsler om feilinformasjon kan vi gjøre endringen umiddelbart om det gjelder nettsiden, men spillet er lokalt på maskinen så vi har liten påvirkningskraft til endre utgitte versjoner. |
| 4 | Spillet har en negativ påvirkning på spillerens mentale helse eller velvære | 3 | B | D | D | C | D | Avhengighet er en stor risiko blant dataspillere. Om vi får mange tilbakemeldinger gjennom nettsiden som varsler om dette, må vi midlertidig stoppe spillet til vi har kommet med en løsning. |
| 5 | Spillet/nettsiden inneholder støtende eller diskriminerende innhold | 2 | A | C | A | D | C | Vi kan endre dette raskt på nettsiden, men kan ikke gjøre annet enn å varsle til de som allerede har lastet ned spillet |
| 6 | Nettsiden virker ikke eller er veldig treig | 4 | A | A | A | B | A | Vi bør stenge serveren midlertidig for å finne årsaken. Kan det være DDOS? |

|  |  |  |
| --- | --- | --- |
| **Sannsynlighet:**  1. Svært liten  2. Liten  3. Middels  4. Stor  5. Svært stor | **Konsekvens:**  A. Svært liten  B. Liten  C. Moderat  D. Alvorlig  E. Svært alvorlig | **Risikoverdi (beregnes hver for seg):**  Menneske = Sannsynlighet x Konsekvens Menneske  Ytre miljø = Sannsynlighet x Konsekvens Ytre miljø  Økonomi/materiell = Sannsynlighet x Konsekvens Øk/materiell  Omdømme = Sannsynlighet x Konsekvens Omdømme |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Svært liten  1 | Liten  2 | Middels  3 | Stor  4 | Svært stor  5 |
| 1 gang pr 50 år eller sjeldnere | 1 gang pr 10 år eller sjeldnere | 1 gang pr år eller sjeldnere | 1 gang pr måned eller sjeldnere | Skjer ukentlig |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gradering | Menneske | Ytre miljø  Vann, jord og luft | Øk/materiell | Omdømme |
| E  Svært Alvorlig | Død | Svært langvarig og ikke reversibel skade | Drifts- eller aktivitetsstans >1 år | Troverdighet og respekt betydelig og varig svekket |
| D  Alvorlig | Alvorlig personskade.  Mulig uførhet | Langvarig skade. Lang restitusjonstid | Driftsstans > ½ år  Aktivitetsstans i opp til 1 år | Troverdighet og respekt betydelig svekket |
| C  Moderat | Alvorlig personskade. | Mindre skade og lang restitusjonstid | Drifts- eller aktivitetsstans < 1 mnd | Troverdighet og respekt svekket |
| B  Liten | Skade som krever medisinsk behandling | Mindre skade og kort restitusjonstid | Drifts- eller aktivitetsstans < 1 uke | Negativ påvirkning på troverdighet og respekt |
| A  Svært liten | Skade som krever førstehjelp | Ubetydelig skade og kort restitusjonstid | Drifts- eller aktivitetsstans < 1dag | Liten påvirkning på troverdighet og respekt |

Det er litt vanskelig å finne alvorlige problemer med et spill på denne skalaen, så jeg har valgt å se for meg at spillet er blitt litt større og at vi har et par tusen brukere.

## Forebygging av datatap

Jeg har all kode og alle dokumenter på GitHub, skole-pc, hjemme-pc og en eldre versjon av prosjektet på en harddrive hjemme. Jeg tar back-up av databasen hver fredag og etter store endringer. Jeg har ikke lagt inn en automatisk backup, men det vil være en naturlig ting å gjøre dersom jeg skal utvikle spillet videre.

# Egenevaluering

**Hvilke utfordringer har du møtt på underveis og hvordan har du løst det?**

En utfordring jeg hadde var da jeg byttet fra en MariaDB database til en PostgreSQL database. De er begge SQL databaser, men har litt forskjellig syntaks. Syntaksen til Postgres er litt strengere. Du må blant annet ha doble hermetegn rundt hvert navn som har en stor bokstav i seg, for at ikke alt skal bli oversatt til små bokstaver når koden kjører. Du må også skrive et enkelt hermetegn når du setter verdier. Dette hadde jeg ikke lest noe sted, men skjønte fort at enkelt hermetegn var til verdier, og at dobbelt hermetegn var viktig, men jeg forsto først ikke helt når jeg skulle bruke dem. Jeg oppdaget at det var mine tabell- og kolonnenavn som trengte dobbelt hermetegn (fordi noen av dem hadde store bokstaver). Så på et punkt satt jeg alle navn i hermetegn. Etter litt mer eksperimentering har jeg nå skjønt når jeg må bruke hva.

**Hva har du lært av å jobbe med årsoppgaven?**

Jeg har lært hvordan man setter opp en server pc, hva objekter er i et kodespråk, og hvilke lover som er relevante om man vil ha en egen nettside eller gi ut noe på nett. Det har både vært interessant og gøy å se hvordan prosjektet mitt gikk fra et lite spill i konsollvinduet til hva det har blitt til i dag.

**Hvordan vurderer du det faglige arbeidet du har gjort?**

Jeg tror jeg har gjort mye bra. Jeg valgte et operativsystem uten desktop for å tvinge meg ut av komfortsonen og lære terminalen og hvordan den fungerer. Jeg valgte en database ingen andre hadde valgt for å utfordre meg selv. Jeg lagde en semi-komplisert nettside med flere funksjoner som ikke var nevnt i kriteriene, selv om det egentlig er spillet jeg skulle vise utviklingskompetanse med. Og jeg har laget fire versjoner av java-spillet ettersom jeg lærte mer, og kom med nye ideer som fikk meg til å utvide spillet i så stor grad at det var enklere å lage et nytt med inspirasjon fra det gamle.

**Hva ville du gjort annerledes om du skulle gjøre årsoppgaven på nytt?**

Jeg har jo lært mye gjennom årsoppgaven og tror om jeg hadde gjort det igjen ville det gått mye raskere nå som jeg kan det jeg trenger å kunne. Noe jeg kunne vært bedre på er å skrive opp tanker og idéer underveis, så jeg ikke risikerer å glemme dem. Jeg får fort høye ambisjoner og finner mange måter å utvide prosjektet på, men jeg risikerer å glemme noen av dem dersom jeg ikke har det skriftlig.

# Kilder

<https://lovdata.no/>: Her hentet jeg lovene jeg syntes var relevante til oppgaven min

<https://stackoverflow.com/>: Jeg lette her når hadde generelle syntax-problemer.

<https://www.geeksforgeeks.org/>: Har også mange gode introduksjons sider for funksjoner

<https://www.w3schools.com/>: Her lærte jeg mye grunnleggende kunnskap samt enkel sql syntax

<https://www.youtube.com/RyiSnow/>: Jeg lærte om Java bibliotekene java.atw og javax.swing av YouTube kanalen RyiSnow i hans tutoriel

<https://www.php.net/>: brukte jeg når jeg bytta til postgres og måtte finne nye php funksjoner får å komunisere med databasen.

<https://i.ntnu.no/documents/>: hadde et risikovurderings skjema jeg brukte, med små endringer.

Jeg har også brukt diverse filer fra OneDrive undervisningsmateriell mappen fra skolen.