



Høgskulen
på Vestlandet

BACHELOROPPGAVE

Adgangskontroll for anleggsområde

Access control for construction site

**Vegard Skjerdal, Henry Lunde Mjåtvedt,
Rune Anseth**

Ingeniørfag - elektro, automatiseringsteknikk
Fakultet for ingeniør- og naturvitenskap/ Institutt
for datateknologi, elektroteknologi og realfag

Veiledere: Marcin Andrzej Fojcik, Olav Sande
25.05.2020

Jeg bekrefter at arbeidet er selvstendig utarbeidet, og at referanser/kildehenvisninger til alle kilder som er brukt i arbeidet er oppgitt, jf. Forskrift om studium og eksamen ved Høgskulen på Vestlandet, § 12-1.

STUDENTRAPPORT

Campus Førde, Svanehaugsvegen 1, 6812 FØRDE www.hvl.no

TITTEL H02-300 Bacheloroppgåve	RAPPORTNR.	DATO 25.05.2020
PROSJEKTTITTEL Adgangskontroll for anleggsområde	TILGJENGELIG Åpen	ANTALL SIDER 70
FORFATTERE Rune Anseth, Vegard Skjerdal, Henry Lunde Mjåtvedt	ANSVARLIGE VEILEDERE Olav Sande, Marcin Andrzej Fojcik	
OPPDRAGSGIVER Proresult AS, Bjørn Erik Kringstad		
SAMANDRAG Oppdragsgiveren for denne bacheloroppgaven er Proresult AS. Proresult har utviklet et datasystem for å registrere brukere inn eller ut av et anleggsområde ved å skanne et HMS-kort på en kortleser. Proresult ønsket å utvide dette produktet slik at en rotasjonsgrind blir styrt av et blåttannrele. Dette vil si at brukere må skanne et gyldig HMS-kort for å gå igjennom rotasjonsgrinden. Oppgaven til Bachelorstudentene var å finne et passende blåttannrele som kan styre grinden, og utvikle en applikasjon som kan styre releet. Releet skulle også innkapsles slik at releet blir beskyttet mot vær. I tillegg til Proresult sin oppgave ble ekstraoppgaver jobbet med samtidig, slik at arbeidsmengden ble bedre tilpasset en gruppe på tre. En klientapplikasjon ble utviklet som gjør det mulig for brukerne å bruke en privat smarttelefon for registrering via NFC istedenfor å bruke kortleseren som er montert ved rotasjonsgrinden. For å holde oversikt over brukerne ble en database opprettet og en applikasjon ble utviklet for å administrere databasen.		
SUMMARY The client for this bachelor thesis is Proresult AS. Proresult has developed a computer system to register users in or out of a work site by scanning an HSE card on a card reader. Proresult wanted to extend this product so that a turnstile is controlled by a Bluetooth relay. This means that users must scan a valid HSE card to go through the rotation gate. The task of the undergraduate students was to find a suitable Bluetooth relay that can control the gate and develop an application that can control the relay. The relay should also be encapsulated so that the relay is protected from weather. In addition to Proresult's task, additional tasks were worked on at the same time, so that the workload was better adapted to a group of three. A client application was developed that allows users to use a private smartphone for registration via NFC instead of using the card reader mounted at the turnstile. To keep track of users, a database was created and an application was developed to manage the database.		
EMNEORD Bluetooth, rotasjonsgrind, rele, Android, database, NFC, HMS-kort		

Forord

Denne rapporten er et resultat fra bacheloroppgave (HO2-300) av tre studenter som gikk 6. semester på bachelor automasjonsingeniør ved Høgskulen på Vestlandet i Førde. Oppgaven teller 20 studiepoeng og det forventes at hver student arbeider cirka 500 timer/student på prosjektet.

Proresult tilbydde denne bacheloroppgaven til høyskolen. Vi takket ja til oppgaven fordi oppgaven var relevant for automasjonsingeniører, og at vi så på det som en fordel å gå for en bedriftsoppgave siden det er relevant for arbeidslivet.

Våren 2020 var en spesiell tid vi alle kommer til å huske da Covid-19 plutselig gjorde det mere utfordrende å møtes og samarbeide.

En stor takk Bjørn Erik Kringstad og Jostein Stuhaug fra Proresult AS for at vi en utfordrede og meningsfull oppgave, i tillegg til tilgangen på produktet deres som gjorde det lettvis å føre arbeidslogg under prosjektperioden.

Takk til veilederne Olav Sande og Marcin Andrzej Fojcik for den gode hjelpen og støtten vi fikk.

Til slutt vil vi takke Joar Sande for koordineringen av bacheloremnet og veiledningen.

Førde 25.05.2020

Vegard, Rune og Henry

Innholdsfortegnelse

Ordliste	7
1. Sammendrag	10
2. Innledning.....	11
2.1. Bakgrunn.....	11
2.2. Mål.....	12
2.2.1. Hovedmål	12
2.2.2. Utvidelsesmål	13
2.2.3. Oppsummering av mål	14
2.2.4. Prioritering av mål.....	15
3. Hoveddel	16
3.1. Oversikt.....	16
3.2. Relekort.....	17
3.2.1. Kravspesifikasjoner for relekort	17
3.2.2. Finne leverandører og komponenter	18
3.2.3. Vurdering relekort og leverandører	19
3.2.4. Innkapsling til relekort.....	20
3.3. Rotasjonsgrind	21
3.3.1. BTX 300 S.....	21
3.3.2. Koble relekort mot rotasjonsgrind	22
3.4. CE-merking.....	28
3.5. Androidutvikling	30
3.5.1. Kortleserenhet.....	30
3.5.2. Android Studio.....	30
3.5.3. Valg av programmeringsspråk.....	31

3.5.4.	Opplæring androidutvikling i prosjektet	31
3.6.	Blåtann.....	33
3.6.1.	Blåtannparing	33
3.6.2.	Nettverkstopologi.....	33
3.6.3.	Blåtannsikkerhet.....	34
3.6.4.	Blåtannsikkerhet i prosjekt.....	35
3.6.5.	Utviklingsprosessen til blåtannapplikasjonene	36
3.6.6.	Implementering av blåtann i applikasjon	37
3.7.	Database	43
3.7.1.	Hva er MySQL	43
3.7.2.	Databasestruktur	43
3.7.3.	Databaseserver.....	44
3.7.4.	Administratorapplikasjon	44
3.8.	NFC.....	47
3.8.1.	Hva er NFC?	47
3.8.2.	Data i NFC	47
3.8.3.	Sikkerhet.....	48
3.8.4.	Hvordan virker NFC	48
3.8.5.	NFC i vår app	49
3.9.	Sammenfletting av blåtann, NFC og database	50
3.10.	Sluttprodukt og resultat.....	51
3.11.	Endringer og avvik.....	55
3.12.	Videreutvikling etter prosjektet.....	56
3.13.	Konklusjon.....	57
4.	Prosjektadministrasjon.....	58

4.1. Organisering.....	58
4.2. Prosjektarbeid og administrasjon.....	60
4.3. Gjennomføring i forhold til planen.....	62
4.4. Økonomi og tidsbruk	63
4.5. Generell prosjektevaluering	65
Referanser	66
Vedlegg.....	70

Figurer

Figur 1: Kortleser - illustrasjon [1].....	12
Figur 2: Blåtannrele [2].....	12
Figur 3: Oversikt over systemet.....	13
Figur 4: Oversikt over system. Denne figuren er et duplikat av figur 3	16
Figur 5: Hensel DK 0606 G [6].....	20
Figur 6: Rotasjonsgrind BTX 300 S [8]	21
Figur 7: Tilpasset koblingsskjema Numato.....	23
Figur 8: Tilpasset koblingsskjema RelayPros. Del av skjema er hentet fra RelayPros.	23
Figur 9: Illustrasjon åpen kollektorutgang	25
Figur 10: Spenningsdeler.....	27
Figur 11: Famoco fx100. [14].....	30
Figur 12: Overblikk Android studio og emulator.....	30
Figur 13 Oversikt over blåtannkommunikasjon	33
Figur 14: Illustrasjon av et typisk piconet [19]	34
Figur 15: Nettverkstopologien for blåtannkommunikasjonen i prosjektet	34
Figur 16: Bilde av hc-05 modulen [21]	36

Figur 17: Flytskjema over hvordan klientapplikasjonen bruker blåttann	39
Figur 18: Flytskjema av "ConnectedThread" klassen	41
Figur 19: Flytskjema over "MainActivity" klassen	41
Figur 20: Flytskjema av "AcceptThread" klassen	42
Figur 21: Tabellen «arbeidere» med eksempeldata	43
Figur 22: SQL skript for å opprette og utforme databasen i prosjektet.....	44
Figur 23: Utskrift av tabell «arbeidere» med klient i NetBeans.....	44
Figur 24: hovedskjerm, status, og ny bruker fra administratorapplikasjon.....	45
Figur 25: Logo for NFC [37].....	47
Figur 26: Illustrasjon av aktiv og passiv NFC-enhet [33]	48
Figur 27: Flytskjema over hele systemet.....	51
Figur 28: Flytskjema som viser hva som skjer når en NFC brikke blir scannet	52
Figur 29: GUI til klientapplikasjon	53
Figur 30: GUI til serverapplikasjon	53
Figur 31: hovedskjerm, status, og ny bruker for serverapplikasjon. Duplikat av Figur 24	54
Figur 32: Organisasjonskart.....	58
Figur 33: Gantt diagram planlagt fremdrift	62
Figur 34: Sektordiagram rapporterte arbeidstimer	64

Tabeller

Tabell 1: Prioritering av mål	15
Tabell 2: Tidsskjema	62
Tabell 3: Regnskap og budsjett	63
Tabell 4: Arbeidstimer planlagt og forbrukt.....	64

Ordliste

AC	Vekselstrøm
Android	Mobilt operativsystem basert på en modifisert linuxkjerne.
Android Studio	Offisielt IDE for utvikling av applikasjoner til Android.
API	«Application Programming Interface». Programmeringsgrensesnitt. Gjør deler av en programvare kan kjøres fra en annen programvare.
AUTO_INCREMENT	Oppretter et unikt siffer i et felt i en tabell når en ny oppføring blir lagt til ved å legge til 1 til det forrige sifferet som ble brukt.
Bluetooth / Blåtann	Trådløs kommunikasjonsprotokoll.
Blåtannrele	Rele styrt via Blåtann
CE-merke	Et merke som viser at produsenten garanterer at produktet oppfyller alle direktiver som har med sikkerhet å gjøre
COM	Felles koblingspunkt i elektrisk krets
DC	Likestrøm
DipSwitch	En eller et sett med vippebrytere for styre enkel funksjonalitet for et kretskort
GPIO	«General purpose input/output». Digital signalpin på et kretskort uten predefinert formål. Bruker kan selv definere om den skal være inngang eller utgang.
GUI	«Graphical User Interface». Grafisk brukergrensesnitt.
HMS	Helse, miljø og sikkerhet.
HMS-kort	Kort som viser hvem den ansatte er og hvilken bedrift personen er ansatt for. Påkrevd for alle som jobber på bygge- og anleggsområder, samt alle som jobber i renholdsbransjen.
HVL	Høgskulen på Vestlandet
IDE	«Integrated Development Environment». Integreert utviklingsmiljø for programmering.
INT	Heltallsvariabel.

IO-kort	Input/output-kort. Integrert krets for digitale eller analoge innganger og utganger
IP-grad	Kapslingsgrad for elektronikk.
Java	Objektorientert programmeringsspråk
Klasse	Koden som bestemmer type data et objekt kan inneholde og hva slags beregninger/operasjoner objektet kan gjennomføre.
Kotlin	Objektorientert programmeringsspråk. Laget for å fungere sammen med Java.
MAC-adresse	«Media Access Control»-adresse. Unik fysisk adresse for utstyr som kan kobles til et nettverk.
MySQL	Et databaseadministrasjonssystem
NCD	National Control Devices. Produsent av elektronikk.
NC	«Normally closed». Koblingspunktet til et rele som danner en lukket krets når spolen i releet ikke er magnetisert. Vil danne en åpen krets når spolen er magnetisert.
NetBeans	IDE for Java.
NFC	«Near field communication». Kommunikasjonsprotokoll for sending eller mottak av data over veldig korte avstander.
NFC-brikke	Passiv elektronisk brikke som kan leses via av en enhet med NFC-teknologi
NO	«Normally open». Koblingspunktet til et rele som danner en åpen krets når spolen i releet ikke er magnetisert. Vil dane en lukket krets når spolen er magnetisert.
Objekt	En del av et kjørende program som inneholder data og kan utføre beregninger og operasjoner på disse.
Pull-upresistans	Resistans i en elektrisk logisk krets brukt til å sikre tilstanden til et signal.
Socket	Internt endepunkt for sending og mottak av data innenfor en node i et datanettverk.
SQL-spørring	Forespørsel om data fra en database

TCP/IP	«Transmission Control Protocol/Internet Protocol». En gruppe kommunikasjonsprotokoller for sammenkobling av datamaskiner i nettverk, for eksempel Internett.
UUID	«Universally Unique IDentifier» 128-bits tall brukt for å identifisere informasjon i datasystemer.
VAC	Vekselspenning
VARCHAR(8)	Tegnvariabel i MySQL med lengde på åtte tegn
WIFI	Trødløs kommunikasjonsteknologi.

1. Sammendrag

Oppdragsgiveren i dette prosjektet er Proresult AS. Bachelorgruppen består av tre personer som studerer automatisering ved Høgskulen på Vestlandet i Førde. Proresult har utviklet et system som kalles Anleggsterminal. Dette systemet brukes for å registrere personell inn og ut på byggeplasser skanning av HMS-kort på en kortleser. Oppgaven har gått ut på å utvide systemet deres med rotasjonsgrind som fysisk barriere. Proresult ønsket at porten skulle bli styrt av et passende blåtannrele, og at releet skulle bli styrt av en androidenhet. Releet skulle også være innkapslet for å beskytte mot vær og andre forhold som støv, siden rotasjonsgrinden skal stå utendørs. Bachelorgruppen undersøkte markedet etter et rele som ble levert med innkapsling inkludert, men det viste seg at dette ikke var mulig. Etter å ha undersøkt mange alternativer ble to relekort bestilt der ene var fra RelayPros og det andre fra Numato Lab. Arbeidsgruppen har konkludert med at et bedre alternativ kan være å bestille et rele i fra en annen leverandør som tilbyr releer som kan tilpasses kundens behov. Grunnen til at dette alternativet ble lagt bort er at leveringstiden var for lang for bacheloroppgaven. Relay Pros sitt rele kom dessverre ikke tidsnok i posten til at gruppen kunne utføre testing på releet. Numato Lab sitt rele kom fort i posten, og testing ble utført. Releet fungerer fint til en prototype, men vil ikke være aktuell i produktet til Proresult. Grunnen til dette er at Numato Lab sitt rele ikke kan konfigureres. Dette betyr at releet ikke kan passordbeskyttes, og gjør det mulig for hvem som helst å koble seg til.

Ekstraoppgaver ble jobbet med parallelt med Proresult sin oppgave. Grunnen til dette er at bachelorgruppen ønsket å ha en større oppgave med flere tekniske utfordringer. I tillegg til å ha en kortleser som styrer blåtannreleet ble en klientapplikasjon utviklet. Klientapplikasjonen blir installert på smarttelefonen til brukeren. Denne applikasjonen kommuniserer med kortleseren via blåtann. Klientapplikasjonen har støtte for NFC som betyr at brukeren kan skanne en NFC-brikke for å åpne porten. En database er også implementert slik at gyldige brukere kan lagres. Hvis brukeren ikke eksisterer i databasen, blir de nektet tilgang. En databaseadministratorapplikasjon ble også utviklet. Denne applikasjonen viser en oversikt over brukere og viser om de er inne eller ute av anleggsområdet. Oppretting eller sletting av brukere er også mulig i denne applikasjonen.

2. Innledning

2.1. Bakgrunn

Bacheloroppgaven ble utgitt av firmaet Proresult AS. De ønsket å videreutvikle det eksisterende produktet sitt Anleggsterminal, som er et datasystem de bruker for å registrere inn og ut personell med HMS-kort. Meningen med Anleggsterminal er bedre sikkerhet og kontroll på arbeidsplassen. I Anleggsterminal blir HMS-kort registrert via en liten androidenhet med en kontaktløs kortleser. Problemet med det allerede eksisterende produktet er at det ikke har noe som fysisk hindrer folk i å gå forbi kontrollen uten gyldig registrering.

Proresult ønsket derfor en utvidelse av Anleggsterminal til å kunne fysisk gi personell adgang eller ikke igjennom rotasjonsgrind. Adgangen skal fortsatt styres via den allerede eksisterende androidenheten, men skal i tillegg styre låsemekanismen til rotasjonsgrind via Blåtann. Til rotasjonsgrind skal det være et relekort med Blåtann som styres fra androidenheten.

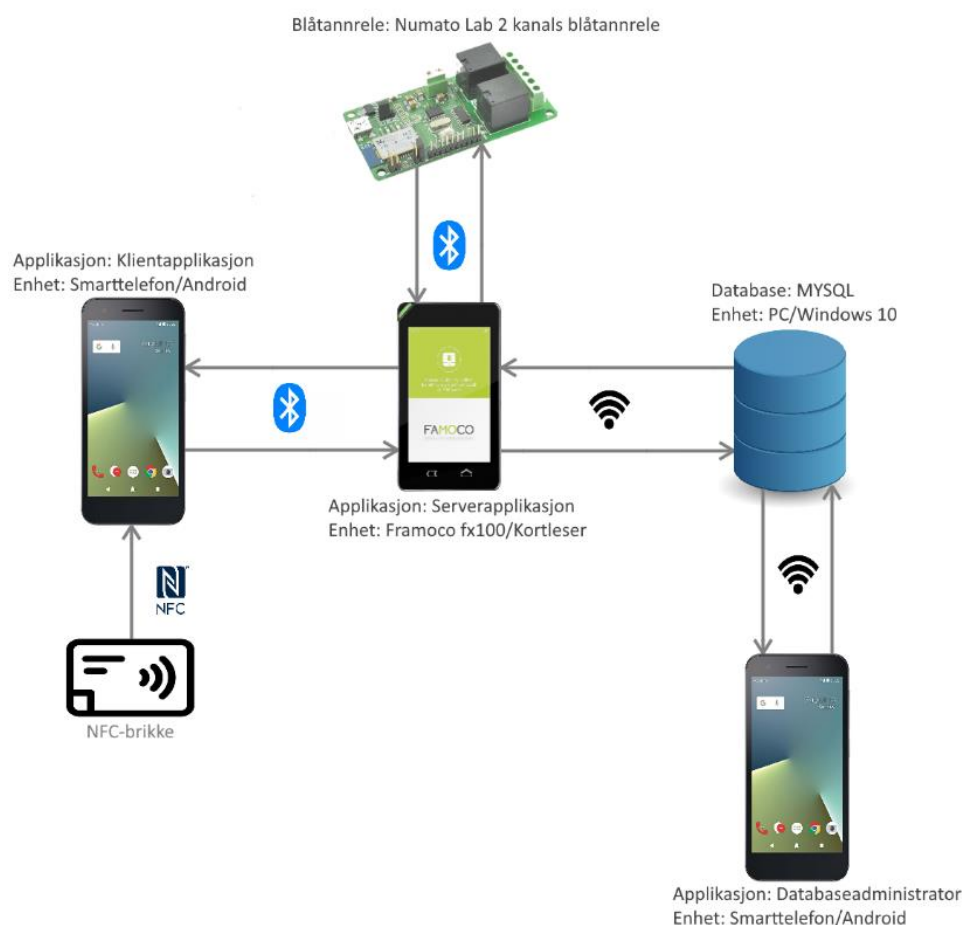
Adgangskontroll er noe som er viktig på et anleggsområde hvor det ligger material og verktøy som kan bli stjålet. Adgangskontroll vil også sikre mot uautorisert personal vandrer inn på farlige områder.

2.2.2. Utvidelsesmål

Ekstraoppgaver skal jobbes med parallelt med Proresult sin problemstilling.

Ekstraoppgavene blir planlagt av gruppen i tilfelle oppgavene fra Proresult har for lite arbeidsmengde i forhold til bacheloroppgave for tre studenter. Ekstraoppgavene har lavere prioritet enn de fra Proresult og er i tillegg uavhengige av Proresult. Her blir det mer fokus på å lage teknisk fungerende prototype enn å lage ferdig produkt.

Applikasjon til Android (telefon) skal utvikles slik at det blir ekstra funksjonalitet. Denne applikasjonen skal kunne kobles til kortleseren og vil dermed oppføre seg som en klient for kortleseren. Applikasjonen vil ta i bruk NFC som gjør det mulig for brukere å verifisere HMS-kort, og åpne porten. En administrerende applikasjon skal også bli utviklet for å vise mannskapsliste. Denne listen skal vise en oversikt over arbeiderne, og indikere om de er inne eller ute av området. Applikasjonen skal også brukes for å legge til nye brukere, eller slette dem. Dette betyr at en database må inkluderes for å hente og lagre data. Databaseserveren skal kjøres på en Windows PC og er av typen MySQL. MySQL ble valgt fordi det er enkelt å bruke og gratis. Grensesnittet mot databaseklientene skal gå via SQL-spørringer. SQL-spørringene blir implementert i kortleseren. En oversikt over prosjektet kan observeres i Figur 3. Figuren viser en oversikt over hvordan klienten, kortleseren, databaseadministratoren og databasen skal settes opp.



Figur 3: Oversikt over systemet

2.2.3. Oppsummering av mål

Hovedmål

Hovedmålet med bacheloroppgaven er å videreutvikle adgangskontrollsystemet til Proresult. Spesifikt er målet å gjøre systemet deres trådløst mellom kortleseren og releet som styrer rondellporten. Som gruppe er målet bli å lage en god løsning som dekker Proresult sine krav og behov.

Effekt mål

Utvikle et godt produkt som er attraktivt for Proresult og kundene deres.

Resultatmål

Utvikle et robust produkt som Proresult kan bruke framover. Det betyr at leverandøren for bluetoothreleet må være stabil, og at releet skal være i produksjon langt fram i tid. Dette gjelder også eventuelle andre komponenter som skulle inkluderes i produktet. Koden som skulle implementeres i androidenheten bør være mest mulig feilfri, lett å lese, og robust.

Delmål

- Lære å programmere applikasjoner til Android sitt operativsystem.
- Lære å bruke Bluetooth API til å sende data til andre enheter.
- Lære å utføre realistiske prosjekt i arbeidslivet.
- Lære å samarbeide med en bedrift for å løse en oppgave.
- Skrive en god sluttrapport.
- Lære hvordan NFC kan brukes i en applikasjon for å verifisere brukere.
- Lære å sette opp en androidenhet som kan kommunisere med en database over TCP/IP.

Reviderte mål

Se kapittelet endringer og avvik (Side 55).

2.2.4. Prioritering av mål

Oppgaver med prioritet 5 og 4 er arbeidskrav, mens fra og med 3 og under er valgfrie oppgaver.

Tabell 1: Prioritering av mål

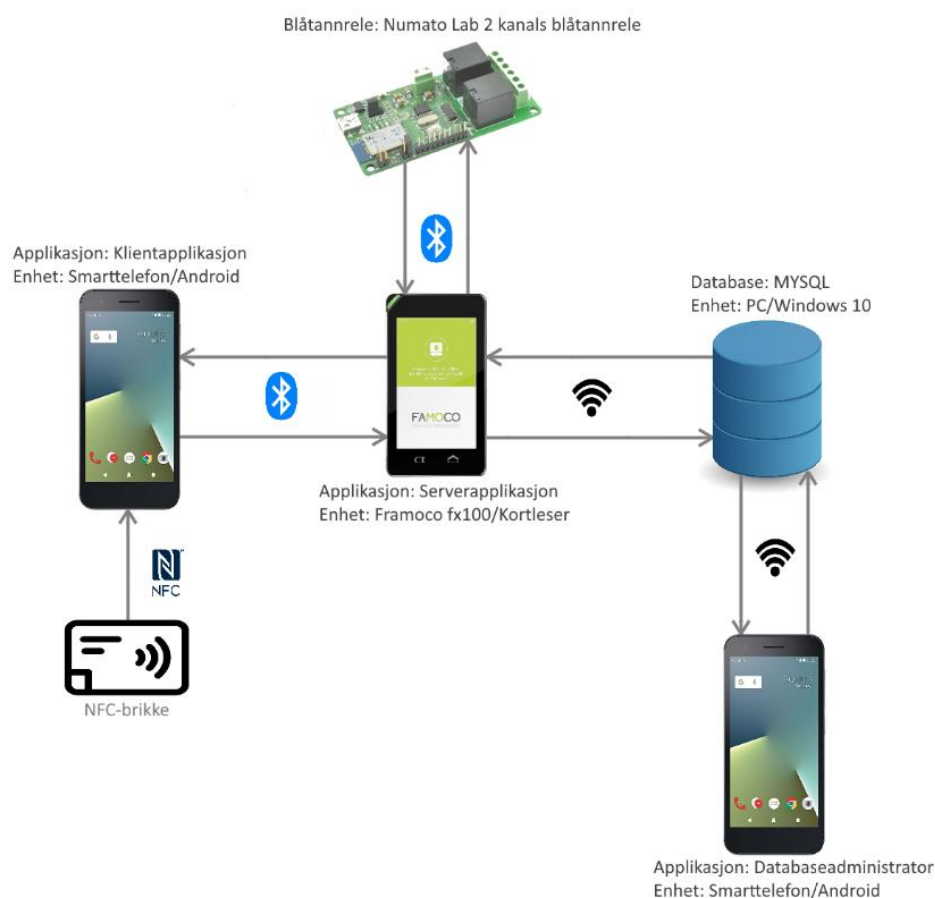
Oppgaver	Rangering (1-5)
Finne en stabil leverandør som selger et innkapslet blåttannrele. Relet må være i produksjon langt fremover i tid.	5
Utvikle applikasjon til Proresult sin kortleser som kan åpne og lukke port ved å kommunisere med blåttannrele.	5
Utvikle applikasjon til Android mobiltelefon som kan kommunisere med kortleseren og åpne porten.	4
Implementere mannskapsliste i applikasjonen, og finne en databaseløsning for lagring og henting av informasjon	3
Implementere støtte for NFC slik at HMS-kort kan leses på mobiltelefon	3

3. Hoveddel

3.1. Oversikt

I kapittel 3 skal alle de forskjellige funksjonalitetene og teknologiene som er brukt i prosjektet forklares. Det kan være greit å ha en oversikt over helheten av oppgaven før en går dypere ned i de forskjellige temaene. Merk at kapittel 3.6 er mer teknisk forklart. Grunnen til dette er at Proresult AS er spesielt interessert i blåtanndelen av prosjektet.

I prosjektet er det altså et blåtannrele som skal styre låsefunksjonen til en rotasjonsgrind. Releet leser også inn et signal i fra porten som kalles «Busy signal». Dette signalet indikerer om porten er opptatt eller ikke. Kortleseren som er montert ved porten kommuniserer med releet gjennom Blåtann, og har mulighet til å sende kommandoer til releet. Arbeidere som har private smarttelefoner, har muligheten til å koble mobilen sinn til kortleseren ved bruk av blåtann. Dette betyr at brukere kan bruke sin egen mobiltelefon til å styre releet/porten. Hvis en bruker ønsker å dra gjennom porten må brukeren skanne en NFC-brikke. Alle gyldige brukere er registrert i MYSQL-databasen, slik at ikke hvem som helst kan gå inn. En applikasjon som oppfører seg som databaseadministrator ble også utviklet for å ha en oversikt over arbeiderne. Applikasjonen kan også brukes for å opprette nye brukere, eller slette dem.



Figur 4: Oversikt over system. Denne figuren er et duplikat av figur 3

3.2. Relekort

3.2.1. Kravspesifikasjoner for relekort

Informasjonen arbeidsgruppen fikk fra oppdragsgiver var i hovedsak at skulle finne et rele som kan styres trådløst med en androidenhet over Blåtann. Releet skal også være tilstrekkelig for å styre låsefunksjonen for rotasjonsgrinden.

Spesifikasjoner for rele/leverandør:

- Blåtann (som kommuniserer med androidkortleser).
- Rele som kan styre låsefunksjonen.
- Vann- og støvbestandig innkapsling (ønske om at releet skulle leveres ferdig innkapslet fra leverandør).
- Fordel med norsk leverandør/importør.
- Kortet måtte kunne styres med egenutviklet app (ikke lukka system som er låst til applikasjon fra leverandør).
- Fungere med rotasjonsgrind BTX 300 S.
- Industriell kvalitet.
- Skal helst kunne kjøpes som en komponent (ikke sammensatt av flere komponenter).
- Stabil og god leverandør som leverer i fremtiden.
- Leverandør måtte ha hurtig nok leveringstid slik at løsningen kunne bli ferdig innenfor prosjektperioden.
- Skal finnes løsning for strømforsyning/kabling til komponent.
- Relekortet skal helst ha en inngang som kan lese opptattsignal fra rotasjonsgrinden.

3.2.2. Finne leverandører og komponenter

Proresult ønsket at det skulle gjøres grundig undersøkelse for å finne produkt og leverandør som passet best til formålet. Gruppen søkte på nettet og fant ganske raskt noen som kanskje ville fungere. Det finest også mye billige kort til noen få dollar, men dette var gruppen enige om å unngå på grunn av kvalitet.

De fleste relekort gruppen fant på nettet som kunne være aktuelle var uten innkapsling og fra utalandsk leverandør. Gruppen lette etter norske leverandører, og fant noe som kanskje kunne brukes til prosjektet som egentlig var ment for låssystemer og noe for elektro. Etter litt nærmere undersøkelse av disse produktene etter at det ble kontaktet viste det seg at de ikke var ikke åpent for å lage egenutviklet applikasjon til de løsningene, men heller låst til en tilhørende applikasjon fra produsent. Det ble spurt noen leverandører om de hadde produkt for oss eller om de visste om andre som hadde, da ble gruppen henvist noe som var uaktuelle til prosjektet. Gruppen konkluderte derfor med at det ikke ble mulig å finne løsning fra norsk leverandør og at det heller ikke var mulig å finne noen som leverte relekort med innkapsling i henhold til kravene. Det ble da konkludert med at innkapsling og elektronikk måtte kjøpes separat.

På et styringsmøte ble Proresult og arbeidsgruppen enig i at det var akseptabelt å kjøpe elektronikk og innkapsling separat, og at det var akseptabelt med utenlandsk leverandør. Det var også enighet i at det kunne være lurt å kjøpe flere forskjellige kort til prosjektet for å teste ut hvilken som kan passe best.

Etter dette møtet jobbet gruppen med lage en skriftlig vurdering og anbefaling utvalgte kort slik at Proresult kunne gjøre en vurdering av hva som skulle kjøpes inn. Denne inneholdt 3 stykk kort der to av dem var spesielt anbefalt å kjøpes. Proresult bestilte to av de kortene som ble anbefalt.

3.2.3. Vurdering relekort og leverandører

Nedenfor er det listet opp noen av de alternativene inkludert drøfting av leverandører/komponenter gruppen fant før det ble kjøpt. Numato Systems BTRL002 er det kortet som ble testet og brukt i prosjektet. Se nedenfor for mere detaljer om relekortene.

Relay Pros R15PL_ZBT (Anbefalt og kjøpt)

Relaypros er leverandør av relekort i USA/Canada. Vurdert til det mest aktuelle alternativet fordi det er kort med de nødvendige funksjonene. Har 1-kanals rele, Blåtann og 8 analoge innganger. Produsenten har mye tilgjengelig dokumentasjon for produktet. Det er mulig å konfigurere blåtannpassord og -ID med hjelp av en tilleggsmodul. Prisen på kortet er noe høy, men dette er trolig på grunn av høy kvalitet. Dette kortet ble anbefalt til Proresult som det mest aktuelle alternativet. Kortet ble bestilt, men dessverre kom leveransen ikke frem selv etter RelayPros hadde sendt på nytt. [2]

Numato Systems BTRL002 (Anbefalt og kjøpt)

Numato Systems er indisk/amerikansk leverandør komponenter til industri/automatisering. Nettbutikken har mange typer relekort, og et aktuelt kort som ble funnet der var et to kanals relekort med blåtann og noen digitale innganger/utganger. Kortet ble anbefalt kjøpt inn fordi det hadde lav pris og nødvendige funksjoner. Ulemper var det ikke ble funnet i dokumentasjonen hvordan konfigurere passord for blåtannstilkobling. For dette kortet var det nødvendig med egen pull-upresistans om en skulle bruke opptattsignalet. [3]

Danbit DIOT-REL2 (Anbefalt)

Danbit er norsk/dansk leverandør. De hadde relekort med to kanaler. Må ha Blåtann/Wifi modul som tillegg. Ulemper er at kortet ikke har innganger. Kortet ble blant de anbefalte kortene, men fikk ikke spesiell kjøpsanbefaling. [4]

Intin (Lagt bort)

Sendt mail for å spørre om de hadde produkt for oss, eller om de visste om noen som kunne ha noe for oss. Fikk svar at Danalock universalmodul med rele kunne muligens passe, og i tillegg tipset om Nordic Semiconductor som spesialiserer seg på kort med Blåtann. Danalock modulen viste seg at hvis en skulle programmere egen app til dette så måtte Proresult bli partner av Danalock, derfor ble dette alternativet lagt bort.

Holars AS (Lagt bort)

Norsk leverandør spesialisert på låssystemer og sikkerhet. De hadde et produkt for låssystem med vanntett innkapsling som så ut til å kunne passe (BT1) [5], men etter å ha kontaktet firmaet fant gruppen ut at en ikke kunne bruke egenutviklet app på denne, derfor ble dette alternativet lagt bort.

Nordic semiconductor (Lagt bort)

Intin tipset gruppen om denne bedriften. Det ble forsøkt å kontakte de for å høre om de hadde passende produkt, men fikk ingen svar. De ble lagt bort som alternativ.

National Control Devices (Lagt bort)

Leverandør i USA som leverer relekort for industriell bruk på bestilling. De kan tilpasse kort etter kundens behov. Gruppen la denne leverandøren bort på grunn av lang leveringstid.

3.2.4. Innkapsling til relekort

Et av kravene til Proresult AS var at produktet måtte ha innkapsling for beskyttelse mot vann og støv. Siden ingen av releene kom med innkapsling fra produsent som dekte disse kravene ble det funnet en leverandør med lokal tilknytning, Ahlsell.

Innkapslingen som ble valgt er av merket Hensel og modell DK 0606 G. Denne modellen ble valgt fordi den hadde plass til releet, hadde fester som passet sammen med releet samt høy IP-grad. IP-graden er 66 som vil si at den er spylesikker selv ved høyt trykk og støvtett. Dette er viktig siden den da tåler regn, og det kan bli en del støv på byggeplasser. Innkapslingen er også flammehemmende og selvslukkende. Det fulgte også med rekkeklemmer, noe som gjør oppkobling på byggeplass lettere. Målene på boksen er 130x77x130 mm, og maks ledertverrsnitt er 6 mm², som er rikelig for våre formål. [6]



Figur 5: Hensel DK 0606 G [6]

3.3. Rotasjonsgrind

I starten av prosjektet spesifiserte Proresult at rotasjonsgrinden som skulle brukes i prosjektet var av typen «BTX 300 S». Denne rotasjonsgrinden produseres av Özak Geçiş Teknolojileri i Tyrkia. Noen norske leverandører tilbyr denne modellen til spesielt til bruk på byggeplasser.

Özak ble grunnlagt i 1974 og ifølge de er de den første og ledende produsent av adgangssystemer for fotgjengere og kjøretøy i Tyrkia. Produsenten eksporterer produkter til mere enn 80 land (Norge er med på listen). [7]

3.3.1. BTX 300 S

BTX 300 S en rotasjonsgrind i full høyde. Ifølge produsenten er modellen egnet for steder med krav til høy sikkerhet. Grinden kan brukes utendørs. Produsenten tilbyr også tilsvarende modell som er dobbel med to rotor (BTX 300 D), den doble har mere passasjekapasitet og mulighet for passasje i begge retninger samtidig. For full dokumentasjon om BTX 300 S, se vedlegg 6.2.2



Figur 6: Rotasjonsgrind BTX 300 S [8]

Noen spesifikasjoner for BTX 300 S:

- Kraftforsyning: 110/220-230 V 50-60 Hz AC. Effektforbruk maks 20 watt.
- Driftstemperatur: (-20°C) – (+68°C). IP-grad 56, (IP 66 alternativt).
- Styresignal(inngang): jordet signal, kan brukes med en eller to styringer. Innganger er vernet med optokoblere.
- Retningsalternativer:
 - En retning adgangkontrolleres med en styring, mens den andre retningen har fri passasje.
 - Begge retninger er adgangkontrolleres med hver sin styring.
 - Begge retninger adgangkontrolleres med en felles styring.
 - Er mulig å blokkere en retning for enveispassasje.
- LED-indikatorer kan vise grønn pil eller rødt kryss.
- Rotoren opereres med manuell dytting, produsent tilbyr også motorisert alternativ.

- Nødmodus kan aktivere fri passasje i begge retninger. Aktiveres ved feil, eller med signal).
- Utgangssignal: Passasjetibakemelding (rele), opptattsignal (transistor).
- Kapasitet gjennomstrømming (manuell): Maks 60 passeringer/min, Nominelt ca. 18 passeringer/min.
- Produsent tilbyr forskjellige alternativer og tilbehør.

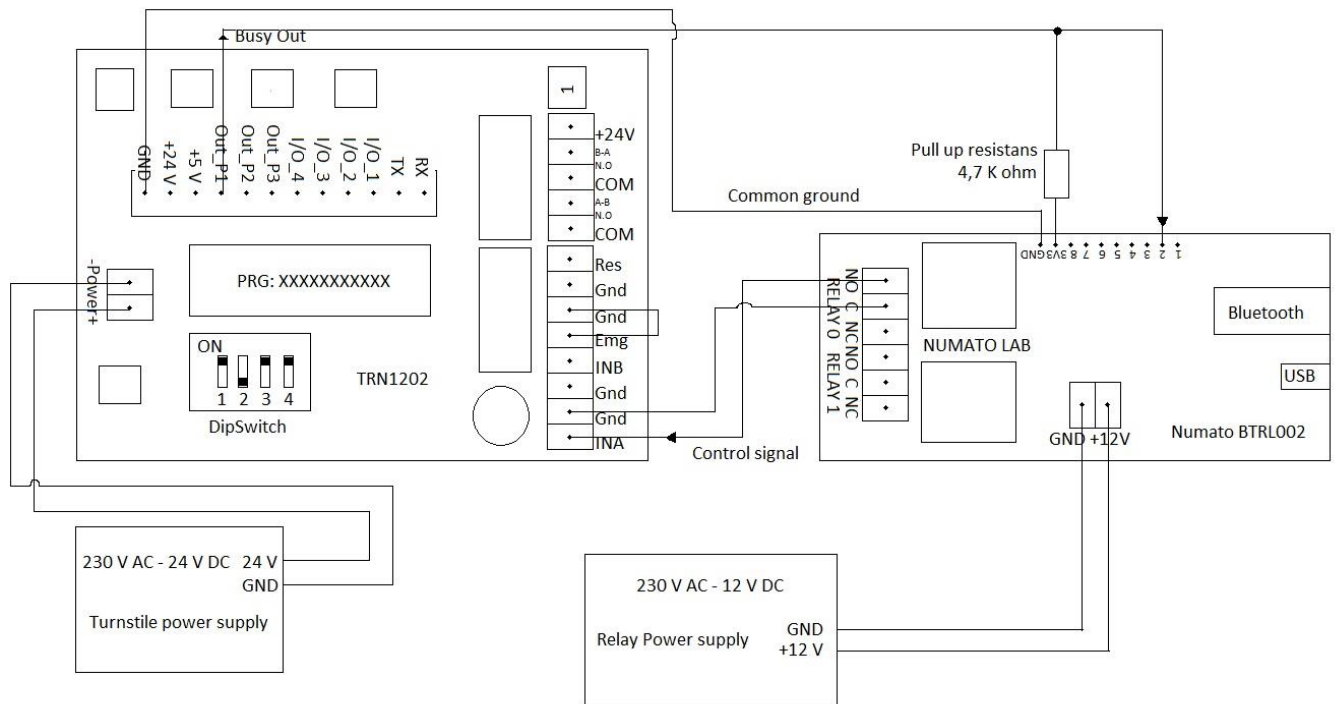
Spesifikasjoner er hentet fra nettsted til produsenten. [8] [9]

3.3.2. Koble relekort mot rotasjonsgrind

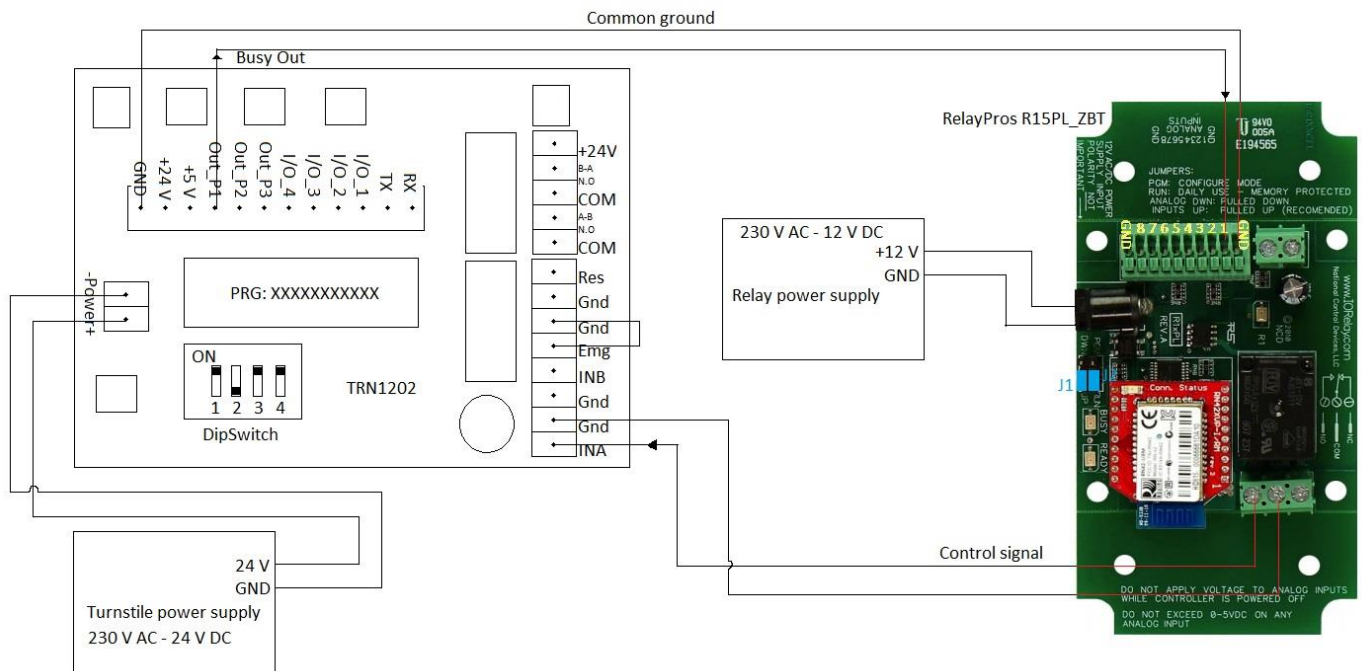
Arbeidsgruppen fikk ikke mulighet til å teste oppkobling på en rotasjonsgrind. En praktisk oppkobling er nødvendig for å bekrefte om løsningen fungerer etter planen. En kan regne med at det vil bli litt prøving og feiling, og at noe må finnes ut av i løpet av testingen. Instruksjonsboken fra produsent kan misforstås, den kan mangle viktige detaljer, og en vet aldri helt hvordan rotasjonsgrinden oppfører seg uten testing. Rotasjonsgrinder kan ha forskjellige tilpassinger for kunder, eller at produsent har endret produktet.

Det ble laget to forskjellige koblings skjema i prosjektet der det ene er tilpasset for Numato-kortet, og det andre er tilpasset RelayPros. Skjemaene som er laget i prosjektet er ment å brukes som et tillegg til produsentens installasjonsguide:

- Tilpasset skjema for kobling med Numato BTRL002 (Figur 7)
- Tilpasset skjema for kobling med Relaypros R15PL_ZBT (Figur 8)
- Produsentens installasjonsguide (Vedlegg 6.2.1)



Figur 7: Tilpasset koblingskjema Numato



Figur 8: Tilpasset koblingskjema RelayPros. Del av skjema er hentet fra RelayPros.

Nedenfor er det forklart deler av skjemaene og hvordan arbeidsgruppen har planlagt å koble relekort til rotasjonsgrinden.

Oppsett for koblingene

Skjemaene er satt opp med kontrollfunksjon at ene gangretning er kontrollert mens den andre retning er fri. Dette er logisk med tanke på at inngang adgangkontrolleres, og at utgang er fri. Det er konfigurert med tidsavbrudd på 12 sekund, dette konfigureres med «DipSwitch». Kraftforsyning til relekort er med batterieliminatort. Merk at det på RelayPros kan det brukes batterieliminatort med nettplugg som plugges i kortet, mens på Numato må lederne festes i koblingsklemmer på kortet. For opptatt signalet brukes det ekstern pullup-resistor på inngang til Numato, i skjemaet er det brukt 4,7 kiloohms motstandsverdi. Verdien på 4,7 kiloohm er valgt fordi dette er en typisk verdi for pullup-motstand. RelayPros kortet har 10 kiloohm pullup-resistor innebygget derfor trengs det ingen ekstern pullup-motstand, dette aktiveres med jumper (J1 i skjema).

Rele

Hvert rele på relekortene har tre tilkoblinger:

- COM (Common): Felles kobling mellom NO og NC.
- NO (Normally open): Kontaktforbindelse når rele er aktivt, ikke kontaktforbindelse når rele er inaktivt.
- NC (Normally closed): Ikke kontaktforbindelse når rele er aktivt, kontaktforbindelse når rele er inaktivt.

Styringskort Rotasjonsgrind (TRN1202)

Hovedkortet til rotasjonsgrinden.

Signal for låsefunksjon (Control Signal)

Styresignalet kommer fra relekortet, rotasjonsgrinden gir adgang ved aktivt signal. Styresignalet fra relekort skal være jordet, ved aktivt signal er det kontakt mellom inngang og jord (INA og GND i koblingsskjema). Skjemaet i installasjonsguiden for rotasjonsgrinden er det illustrert at signalet inn kan styres av rele, eller trykknapp eller transistor med et signal. Det er spesifisert fra produsent at styresignalet skal ha varighet på 0,5 -1 sekund.

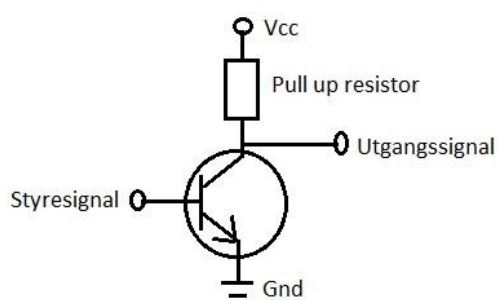
Nødmodus (Emg)

Ved aktiv funksjon er rotasjonsgrind åpen for fri adgang i begge retninger, aktiveres ved åpen kontakt (NC), eller systemfeil. I de tilpassede skjemaene er denne koblet direkte til jord (GND). I eldre modeller av rotasjonsgrind aktiveres nødmodus ved lukket kontakt (NO). Noen rotasjonsgrinder er tilpasset slik at rotasjonsgrind er låst nødmodus i stedet for fri.

Opptattsignal (Busy Out)

Det er en stor fordel å bruke utgang for opptatt signalet for passasje («Busy Out») fordi varighet på tiden for en passering er varierende, og at styringen ikke vet om noen passerer ut. Alternativt uten opptattsignalet kan det programmeres tidskonstant for tid mellom hver opplåsing. Ulempen uten opptattsignalet er at systemet vil bli asynkront med passeringene i rotasjonsgrinden, som kan føre til mindre tidseffektivt system, eller at folk må registrere flere ganger for å komme seg inn.

Utgangen for opptattsignalet er en transistorutgang som en «åpen kollektor» utgang (illustrert i Figur 9). Under passasje vil signalet ut ha «lav verdi», mens når det ikke er passasje har utgangssignalet «høy verdi». Det må kobles en pullup-resistor på inngang til relekort med en spenning som er tilpasset til den spenningen inngangen krever (for eksempel 5 Volt). Noen IO-kort har pullup-resistans innebygget, mens andre som ikke har det må ha en egen ekstern resistans. For at det skal være mulig for relekortet å registrere signalet er det nødvendig at begge kortene er koblet sammen med en felles jording (Common ground).



Figur 9: Illustrasjon åpen kollektorutgang

Vippebrytere (DipSwitch)

Brukes til å konfigurere innstillinger for tidsavbrudd («time setup») og funksjonsprogram («Function program selection»). Funksjonsprogrammet er blant annet for hvilken retning som skal være fri og hvilken retning som skal adgangs kontrolleres. Mer om kombinasjoner for vippebryteren forklart i produsentens dokumentasjon på side 13 i vedlegg 6.2.2.

I de tilpassede koblingsskjemaene er vippebryteren konfigurert med:

- Tidsinnstillinger: 12 sekund tidsavbrudd.
- Funksjon: Adgangskontrollert i retning A-B, fri adgang i retning B-A.

Kraftforsyninger

Kraftforsyningen som leveres med rotasjonsgrinden går på 230 Volt AC, og gir ut 24 V DC, maks 50 Watt. Denne forsyner rotasjonsgrinden og dens funksjoner. De aktuelle relekortene krever 12 V DC. Relekortene kan ikke kobles direkte til den tilhørende kraftforsyningen til rotasjonsgrinden. Mulige alternativer til kraftforsyning for relekort:

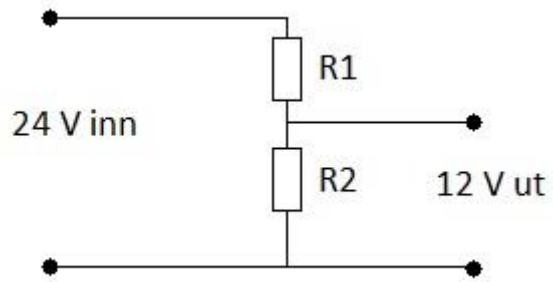
1. Egen kraftforsyning 230 VAC til 12V DC for relekortet (batterieliminatør)
2. Lage spenningsdeler med hjelp av to resistanser til å halvere spenning fra 24 V til 12 V.
3. Bruke spenningsregulator (halvleder).
4. Bruke et tilpasset relekort som kjører på 24 V.

Drøfting av alternativer de ulike alternativene for kraftforsyning:

1. Batterieliminatør kan være et godt alternativ, men krever et ekstra strømuttak. Det kan trolig lages en løsning for nettuttak/forgrening nær kraftforsyningen som allerede er der, men det kommer an på om det er nok plass i koblingsboksen. Det må undersøkes og gjøre en prøvekobling på rotasjonsgrind for å finne ut om dette alternativet er praktisk mulig.
2. Spenningsdeler (illustrert i Figur 10) tar liten plass og kan fungere. Ulempen er at spenningen til relekortet vil kunne variere etter effektforbruket fra relekortet så løsningen kan bli ustabil. Det er ikke oppgitt nøyaktig effektverdi fra produsent, strømforbruket til relekort må derfor måles for å kunne gjøre valg av motstander.
3. Spenningsregulator er en liten og billig komponent (illustrert i **Feil! Fant ikke referanse-kilden.**) som kan regulere ned utgangsspenningen til en fast spenning. Vil trolig fungere godt teknisk, men monteringen krever arbeid med lodding på et eget kretskort.
4. Noen leverandører leverer relekort tilpasset for kunde. Om en får bestilt tilpasset relekort som går på 24 volt så kan det kobles direkte til kraftforsyningen. Dette kan være den praktiske beste løsningen dersom Proresult ønsker en løsning som et salgbart produkt med enklest mulig montering.

Anbefaling valg av strømforsyning: Det anbefales alternativ 4 med tilpasset relekort som går på 24 Volt fordi da vil det bli mye enklere å koble og en kan koble til spenning som allerede er installert. Ellers anbefales alternativ 1 som kan fungere bra som en midlertidig løsning

inntil produktet er videreutviklet. De tilpassede koblingsskjemaene viser kobling med batterieliminatort (alternativ 1).



Figur 10: Spenningsdeler

3.4. CE-merking

Generelt om CE-merking

CE-merket (Communauté Européenne) er produsentens garanti for at produktet tilfredsstillende myndighetenes krav gitt i direktiv eller forordning. CE-merke er et krav for mange produktgrupper som selges. Dette inkluderer blant annet alle elektriske produkter. CE-merking er vanligvis ikke en form for sertifisering eller godkjenning, men heller basert på fabrikantens egenerklæring. Produsentens representant har ansvar for at produktet oppfyller disse kravene, og at dette kan dokumenteres overfor offentlige instanser som kontrollerer produktet. CE-merket er en del av EØS og gjelder for alle EØS-land. Kravene gjelder innenfor helse, miljø og sikkerhet, men ikke kvalitet på produktet. For å bruke CE-merket skal produsenten foreta samsvarsvurdering. Prosedyren for samsvarsvurderingen er angitt i hvert direktiv/forordning. Produsenten skal selv påføre CE-merket etter det er gjennomført samsvarsvurdering. CE-merket skal plasseres på produktet slik at merket er godt synlig, lesbart og det skal være vanskelig å fjernes. En samsvarserklæring skal medfølge produktet. Dokumentasjonen skal produsenten oppbevare i 10 år og skal utleveres ved anmodning fra myndighetene. Identifikasjonsnummeret til kontrollorganet skal påføres. Korrekt merket produkt kan selges fritt i EØS-området. Produkter som ikke er merket, eller feilaktig merket, eller ikke oppfylles av kravene vil bli krevd trukket fra markedet. I Norge er det Direktoratet for samfunnssikkerhet og beredskap (DSB), og Klima- og forurensningsdirektoratet (Klif) som har tilsyn for CE-merking. [10] [11]

CE-merking for produktet i prosjektet

Elektriske produkter er en av produktkategoriene som det er krav om korrekt utført CE-merking for at det skal kunne selges i EU/EØS-område. Dersom Proresult setter sammen et produkt bestående av flere komponenter som deretter skal videreselges i EU/EØS, blir Proresult den ansvarlige produsenten for CE-merking av produktet.

Nedenfor har vi listet opp noen direktiver/forordninger som kan bli gjeldende for den type produkt som i prosjektet:

- Direktiv 2014/35/EU (Lavspenning)
- Direktiv 2014/30/EU (Produkter som kan framkalle elektromagnetiske forstyrrelser)
- Direktiv 2014/53/EU (Radio- og teleterminalutstyr)
- Direktiv 2006/42/EC (Maskiner)
- Direktiv 2006/42/EC (Støy fra utstyr til utendørs bruk)
- Direktiv 2011/65/EU (Farlige stoffer i elektrisk og elektronisk utstyr (RoHS))

Direktivene er hentet fra nettsiden til Standard Norge. [12]

Det er trolig en krevende prosess for Proresult å CE-merke produkt som kan involvere flere direktiv. Det anbefales å forhøre med noen eksterne virksomheter rundt det praktiske og regler for CE-merkingen.

Direktoratet for samfunnssikkerhet og beredskap, og Klima- og forurensningsdirektoratet er de som utfører kontroll av CE-merking i Norge. Nemko er en bedrift som tilbyr tjenester for CE-merking av elektroniske produkter [13]. Standard Norge har en del informasjon på websidene om CE-merking [10].

Alternativt dersom Proresult får kjøpt et ferdig produsert produkt som allerede er CE-merket fra en annen produsent vil ikke Proresult lenger stå som den ansvarlige produsent for CE-merkingen.

3.5. Androidutvikling

3.5.1. Kortleserenhet

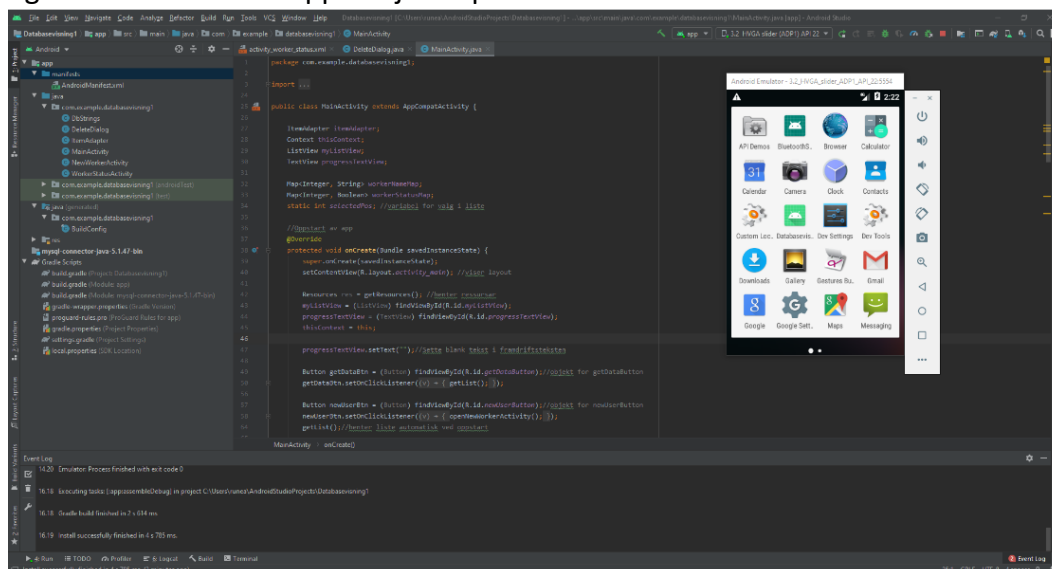
Gruppemedlemmene fikk lånt hver sin kortleserenhet fra Proresult som skulle brukes til testing av android-applikasjoner utviklet i prosjektet. Kortleserenheten som brukes i prosjektet er av typen Famoco FX 100, som er en håndholdt Android-enhet med beregnet til bedriftsbruk (se Figur 11). Enheten har NFC-leser på baksiden, støtte for mobildata, Wifi og Blåtann [14]. Enhetene som arbeidsgruppen bruker, kjører forskjellige eldre versjoner av Android. Å utvikle applikasjoner til enheten gjøres på samme måte som om det var en mobiltelefon. Siden det er eldre Android versjon på enheten måtte gruppen passe på å velge et API i Android studio som var kompatibel enheten. Flere detaljer om enheten kan sees i vedlagt produktark (Vedlegg 6.3).



Figur 11: Famoco fx100. [14]

3.5.2. Android Studio

Android Studio er det offisielle IDEet fra Google til utvikling av Android Applikasjoner. I prosjektet ble det brukt versjon 3.6.3. Kan velges om det skal brukes Java, eller Kotlin som programmeringsspråk. Android studio har en innebygd emulator der en kan teste egenutviklet Android applikasjoner på PC.



Figur 12: Overblikk Android studio og emulator

3.5.3. Valg av programmeringsspråk

Det var et krav fra arbeidsgiver å bruke Java eller Kotlin. Valget mellom disse programmeringsspråkene måtte tas i forprosjektet. Begge er objektorienterte språk, og ligner noe på hverandre.

Kotlin er et kryssplattformsspråk som er laget for å fullt ut med Java og Java Virtual Machine. Første versjon av Kotlin kom ut i 2011. Java kom med sin første versjon i 1995. Java har eksistert lenger enn Kotlin. Dette gjør at det er mer ressurser tilgjengelig for Java enn det er for Kotlin. I tillegg til at det er flere ressurser for Java. Både på skolen og på Internett. Dette er også et språk vi er godt kjent med fra emner som DA2-100 og eget arbeid utført på fritiden og i tidligere prosjekter, som gjør at vi kan få en «running start» på dette prosjektet. Vi må lære oss hvordan utvikle applikasjoner for Android, men slipper å lære oss et nytt programmeringsspråk i tillegg om vi velger Java. [15] [16]

3.5.4. Opplæring androidutvikling i prosjektet

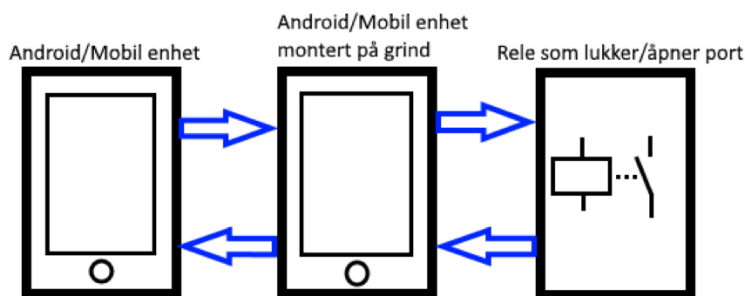
Ved oppstart av prosjektet hadde gruppen så å si ingen erfaringer eller opplæring med å utvikle androidapplikasjoner. En del tid og utfordring med prosjektet ble derfor prosessen å tilegne tilstrekkelige ferdigheter for å utvikle applikasjonene i oppgaven. Planen var at selve opplæringen skulle gruppen gjøre på egenhånd og finne lærestoff selv. På skolen i Førde fantes det ingen lærere som hadde god kompetanse på utvikling av androidapplikasjoner. Det ble undersøkt om det var mulig å få hjelp fra lærer i fra Bergen, men viste seg å ikke være praktisk mulig å få slik hjelp. Gruppen hadde noe erfaring med javaprogrammering fra tidligere opplæring i skolen, derfor regnet gruppen med på forhånd at å begynne å utvikle androidapplikasjoner skulle gå greit, selv om Proresult advarte om at det er omfattende arbeid å utvikle androidapplikasjoner.

Selve opplæringen androidutvikling startet i forprosjektet etter at gruppen fikk utlevert kortleserenhetene fra Proresult. For å komme i gang ble det installert Android Studio på PC, deretter ble det funnet og fulgt noen instruksjoner til veldig enkle applikasjoner på nett som ble overført til og kjørt på kortleserenheten. I løpet av de første timene med Android Studio syntes gruppen det var mye mere annerledes og uvant enn ventet å programmere til tross for erfaringen med Java. Gruppemedlemmene brukte mye tid til å se kurs om androidutvikling på Youtube og se på eksempelapplikasjoner nedlastet fra nett. Det var mye som var vanskelig med å bruke Android studio i starten, som for eksempel å laste inn nedlastede eksempelprosjekt, få overført applikasjoner til kortleserenhet, etter hvert så løste problemene seg med å bruke riktige teknikker. I løpet av prosjektperioden har gruppemedlemmene behersket androidutvikling bedre og bedre etter hvert som gruppemedlemmene måtte finne ut av og bruke nye teknikker. Til slutt ble godt nok til å utvikle applikasjonene i prosjektet.

I prosjektet ble gratis og fritt tilgjengelig stoff på nett brukt som eneste hjelperessurs, gruppen hadde ingen eksterne personer til hjelp eller kostbare bøker. Hadde gruppen valgt Kotlin i stedet for Java som programmeringsspråk ville det ha blitt enda mere nytt å måtte lære og starten hadde ha gått tregere, på den måten var valget av Java det riktige valget av programmeringsspråk.

3.6. Blåtann

I dette prosjektet er trådløs kommunikasjon brukt mellom enhetene for å styre porten. Kommunikasjonsprotokollen som er brukt i prosjektet er blåtann. Grunnen til at blåtann ble brukt er at Proresult AS ønsket å ta i bruk denne protokollen til prosjektet. Figur 13 viser en oversikt over blåtannkommunikasjonen mellom enhetene (andre kommunikasjonsprotokoller er ekskludert i dette kapittelet). Den mobile enheten (kortleseren) som er montert på grinden kommuniserer med både rele og en vilkårlig mobil enhet. Med dette er det ment at en bruker som har applikasjonen installert på mobilen sin kan koble seg til den fastmonterte enheten for å åpne porten. Dette gjør at brukeren enten kan bruke den fastmonterte enheten, eller bruke privat enhet for åpning av port.



Figur 13 Oversikt over blåtannkommunikasjon

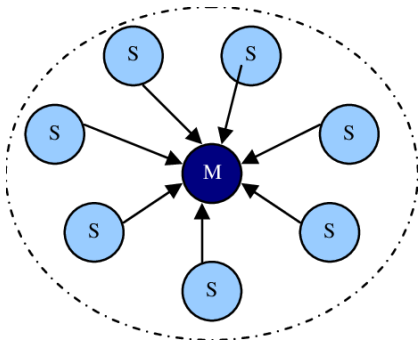
3.6.1. Blåtannparing

For at to eller flere blåtannenheter skal kunne kommunisere med hverandre må de først være «paret». For å pare to enheter må en av enhetene være «discoverable». Dette er en egenskap som kan konfigureres på enheten, og gjør det mulig for andre enheter å oppdage enheter som er «discoverable». Når en bruker prøver å koble seg til en enhet, og mottakeren aksepterer tilkoblingen vil «paringsprosessen» startes. Enhetene vil utveksle sikkerhetsnøkler. Begge enhetene vil få opp en nøkkel på skjermen og hvis disse er like kan brukerne trykke «godta» slik at enhetene blir paret. Denne prosessen trengs bare å utføres en gang ved mindre en av enhetene bryter båndet. Denne paringsprotokollen kalles “simple secure pairing”(SSP). I kapittel 3.6.3 blir SSP forklart i nærmere detalj. Når to enheter er paret, er det enkelt å opprette kontakt for utveksling av data. [17]

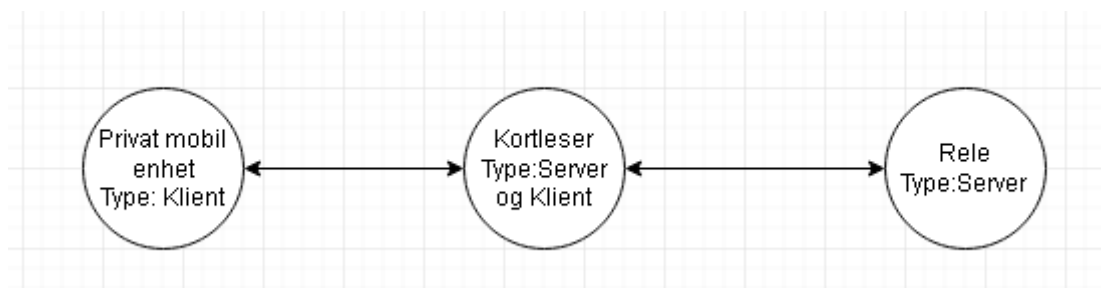
3.6.2. Nettverkstopologi

Nettverkstopologien som er mest brukt for blåtann nettverk er «pico net» (Figur 14). I denne nettverksstrukturen vil en enhet oppføre seg som «master» mens enheter som kobler seg til «masteren» kalles «slaver». I dette prosjektet har blitt enhetene referert som server og klient. Nettverksstrukturen for dette prosjektet kan observeres i Figur 15. Noe som er spesielt med denne strukturen er at kortleseren er av type server og klient. Grunnen til dette er at releet som er brukt til testing ikke kan konfigureres som klient. Dermed må kortleseren

koble seg til releet som klient, samtidig som den må akseptere tilkoblinger ifra andre klienter. [18]



Figur 14: Illustrasjon av et typisk piconet [19]



Figur 15: Nettverkstopologien for blåttannkommunikasjonen i prosjektet

3.6.3. Blåtansikkerhet

De beste sikkerhetstiltakene en bruker kan utføre, er å ikke aktivere blåttann ved mindre det skal brukes. Det er også mulig å gjøre enheten din usynlig for andre blåttannenheter i innstillingene.

Sikkerheten til blåttannprotokollen ligger hovedsakelig i paringsprosessen. Androidenheter bruker en protokoll som kalles SSP (simple secure pairing). Når to enheter skal pares dukker det opp en nøkkel hos begge enhetene. For å akseptere paringen må begge enhetene «trykke» godta, som vil resultere i at enhetene er parert. SSP sørger for at enheter kan pares på en sikker måte. Dataen blir kryptert, og protokollen sørger for at sniffing av paringsprosessen ikke er mulig (eller svært vanskelig). I dette prosjektet brukes SSP for androidenheter. [20]

En annen metode som kan brukes er «link manager protocol» (LMP). LMP er en pinkode basert metode. Denne metoden er ofte brukt hvis en av enhetene ikke har brukergrensesnitt. F.eks. Hodetelefoner, rele eller høyttalere. Denne metoden krever at enheten som kobler til må taste inn en pinkode for å pare med enheten uten grensesnitt. Grunnen til at SSP er vanligst å bruke er at det er sikrere mot «sniffing». Hvis enheter bruker LMC er det lettere å sniffe paringsprosessen som kan føre til at pinkoden kan bli observert. [20]

SSP og LMP er to svært omfattende protokoller, som ikke forklares i detalj i denne rapporten. For å lese mer om disse protokollene kan denne kilden undersøkes: [20].

3.6.4. Blåtannsikkerhet i prosjekt

I prosjektet er det viktig med sikkerhet slik at ikke hvem som helst har tilgang til systemet. Kortleseren som er montert ved porten vil bli satt til «not discoverable». Dette vil si at kortleseren fortsatt kan kommunisere med releet som allerede er parert, og klienter som er parert. Med dette hindres fremmede enheter i å kunne koble til kortleser siden den ikke vil være synlig. Siden dataen som blir sendt mellom enhetene er kryptert er det heller ikke mulig å «sniffe» for å stjele vital data. Det som kan være problematisk med denne løsningen er at hvis nye brukere skal gjennom porten må kortleseren være synlig for nye enheter, slik at nye brukere kan koble seg til med privat mobil enhet. For å løse dette må verifisering og autentisering foregå i et annet nivå. Dette betyr at selve blåtannilkoblingen er åpen, men tilgang til funksjonalitet og data må være beskyttet. Siden Proresult AS skal implementere prosjektet inn i deres system, vil funksjonaliteten og dataen være beskyttet av deres system.

3.6.5. Utviklingsprosessen til blåtannapplikasjonene

I dette kapitlet skal det kort forklares hvilke stadier blåtannapplikasjonene gikk igjennom før den siste versjonen var klar. Se kapittel 3.6.6 for å se en utdypet forklaring av blåtannapplikasjonene.

Det første skrittet mot å utvikle en applikasjon som bruker Blåtann er å lage en veldig enkel applikasjon med minimum funksjonalitet. I starten av semesteret ble en blåtann-modul kjøpt inn som kunne brukes for testing, før releet ankom i posten. Modulen heter hc-05 (se Figur 16). Modulen ble koblet opp mot en Arduino Uno. Den første applikasjonen som ble utviklet kunne koble seg til hc-05 modulen og sende data. Applikasjonen hadde bare to knapper som ble brukt for å sende enere og nuller.



Figur 16: Bilde av hc-05 modulen [21]

Når releet ble tilgjengelig ble den eksisterende applikasjonen bygget videre på slik at den kunne sende kommandoer til releet. Applikasjonen ble også utvidet slik at den kunne lese inn data. Dette ga mulighet for å observere relestatusen (av/på) i applikasjonen. Styringen av releet ble utført med to knapper.

Applikasjonen kunne dermed styre og lese responser fra releet. Denne applikasjonen ble installert i kortleseren. Det neste skrittet var å utvikle en klientapplikasjon som kunne koble seg til kortleseren, og styre releet. Klientapplikasjonen skulle installeres i en smarttelefon. For at dette skulle være mulig, måtte kortleseren oppføre seg som en Blåtannserver (mer om klient og server i kapittel 3.6.6).

Blåtannserveren var satt opp i kortleseren og en enkel klientapplikasjon var utviklet. Dette betydde at en smarttelefon med klientapplikasjonen installert kunne koble seg til kortleseren med blåtann og styre releet.

3.6.6. Implementering av blåtann i applikasjon

For å ta i bruk Blåtann i Android Studio brukes Android Bluetooth API. Dette APIet gir tilgang til mange funksjoner som trengs for å kommunisere med andre enheter trådløst. Målet med dette kapitlet er å gi en forståelse for hvordan APIet kan brukes, og gi en oversikt over hvordan applikasjonen fungerer.

Blåtannklient og Blåtannserver

Hvis to eller flere enheter skal kommunisere gjennom blåtann må en av enhetene oppføre seg som en blåtannserver. De andre enhetene er klienter og har mulighet til å opprette kontakt med en server [22]. I prosjektet er kortleseren satt opp som en blåtannserver, mens den private mobilenheten er satt opp som klient. For å gå dypere inn i hvordan serveren og klienten opererer er det noen viktige begrep og konsepter som må forklares:

Blåtannadapter og «BluetoothDevice»:

Blåtannadapteren er et objekt som representerer den lokale blåtannadapteren til enheten. Adapteren kan utføre handlinger som å aktivere «device discovery», hente ut parede enheter, finne en enhet ved gitt MAC-adresse etc. I dette prosjektet blir adapteren brukt for å returnere parede enheter, og returnere en enhet basert på MAC-adresse. Enhetene som blir returnert blir lagret som en «device» («BluetoothDevice» vil bli referert som «device»). Dette er et objekt som representerer en enhet. [23] [24]

BluetoothSocket:

En «BluetoothSocket» er et objekt i Blåtann APIet som brukes for å opprette en kobling mellom to enheter [25]. For å opprette en «socket» trengs et «device»-objekt som representerer enheten som skal tilkobles. «Socket» objektet gir tilgang til to viktige objekter som kalles «InputStream» og «OutputStream». Når en tilkobling er opprettet kan disse objektene brukes til å sende og motta data. «InputStream» brukes for å lese data som kommer inn gjennom tilkoblingen som er opprettet [26]. Dataen som leses er i form av Bytes. «OutputStream» brukes for å sende data i form av Bytes [27]. Når en «socket» opprettes vil den tilknyttes en unik identifikator, eller ID. Denne identifikatoren kalles «universally unique identifier» (UUID) som er et 128-bit tall [28]. UUIDen skal være helt unik for hver eneste bruker. Grunnen til at det brukes i sammenheng med en «socket» er å garantere at du ikke kobler deg til vilkårlige enheter, og sender/mottar data fra vilkårlige enheter.

ServerSocket:

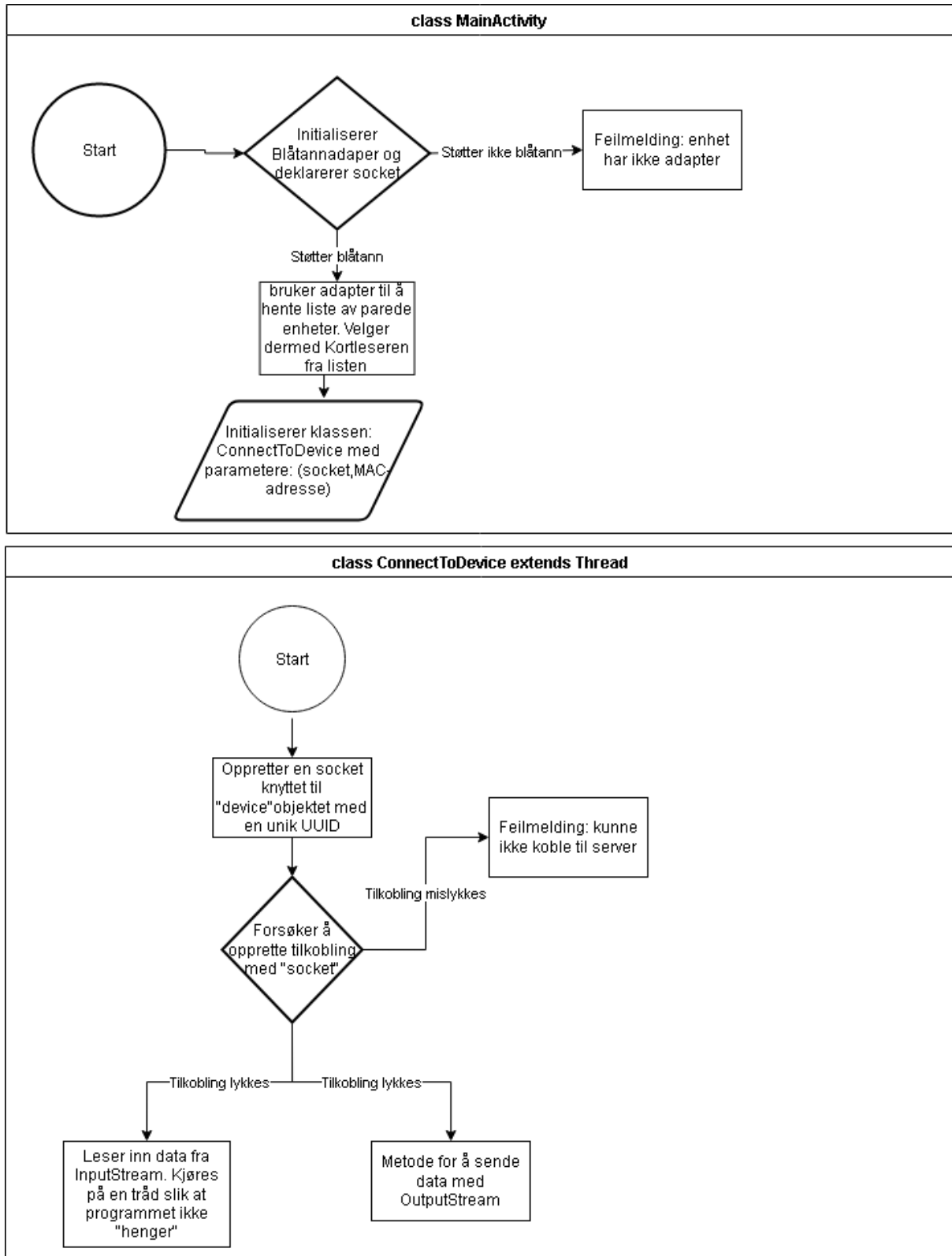
En blåtannserver vil «lytte» etter enheter som ønsker å opprette kontakt. Hvis en klient forsøker å koble seg til serveren og har rett UUID vil serveren akseptere tilkoblingen. «Serversocket» er objektet som lytter etter innkommende tilkoblinger, og når en kobling er akseptert vil «Serversocket» returnere en «socket» som representerer tilkoblingen mellom enheten og serveren. Selv om en enhet er akseptert og en «socket» er opprettet vil «serversocket» fortsette å lytte etter nye tilkoblinger. [29]

Klientapplikasjon

Klientapplikasjonen kan brukes til å åpne porten ved hjelp av en gyldig NFC-brikke. Når brukeren er innen rekkevidde i forhold til kortleseren trengs kun et tastetrykk til for å koble seg til serveren. Når brukeren er tilkoblet er det bare å legge NFC-brikken inntil enheten og porten vil åpnes. Samtidig blir statusen til arbeiderne oppdatert i databasen. Merk at det kun er blåtann funksjonaliteten i applikasjonen som gjennomgås i dette delkapittelet. I Figur 17 er det et flytskjema som viser en oversikt over hvordan applikasjonen fungerer. Figuren kan være nyttig å holde øye med parallelt med forklaringen.

Ved oppstart av klientapplikasjonen blir blåtannadapteren initialisert. Programmet vil søke etter en blåtannadapter på din enhet, og hvis enheten ikke støtter blåtann, vil det komme opp en feilmelding. For å få tilgang til alle parede blåtannenheter brukes blåtannadapter objektet. Disse enhetene blir representert som en liste. Når ønsket enhet er valgt(kortleser/server i dette tilfellet) hentes MAC-adressen til enheten. Nå som en enhet er valgt og MAC-adressen er tilgjengelig kan «ConnectToDevice» klassen initialiseres. Det første som utføres i denne klassen er å få et «device»-objekt som representerer kortleseren. Dette utføres med å opprette en «device» som er knyttet til MAC-adressen. Når et «device»-objekt er initialisert, opprettes en «socket» som representerer tilkoblingen. Når «socket» blir opprettet knyttes den til en unik UUID. Det siste som gjenstår er å bruke «socket» til å starte tilkobling til kortleseren(socket.connect()). Denne kommandoen vil forsøke å opprette kontakt ved gitt UUID. Problemet med kommandoen er at applikasjonen vil fryse. Det betyr at applikasjonen vil fryse frem til en tilkobling er akseptert, eller at en «expection» oppstår. For å fikse dette problemet kjøres denne kommandoen på en separat tråd. Dette gjør at resten av applikasjonen kjøres på en separat tråd og blir derfor ikke påvirket av at socket.connect() fryser.

Når tilkobling er fullført må programmet håndtere data som skal sendes og mottas. For å lese data som blir sendt fra kortleseren brukes en «InputStream» Inputstream opprettes ved hjelp av «socket» som representerer koblingen. For å lese data brukes Inputstream.read(). Denne kommandoen returnerer dataen i form av bytes, og vil henge frem til data mottas eller en «expection» oppstår. Derfor må innlesing også foregå på en separat tråd. For å sende data til kortleseren brukes «OutputStream» som opprettes ved hjelp av «socket». En metode som heter sendMessage(String melding) brukes for å sende data. Metoden tar inn data som parameter og sender data med: OutputStream.write(Bytes bytes)).



Figur 17: Flytskjema over hvordan klientapplikasjonen bruker blåtann

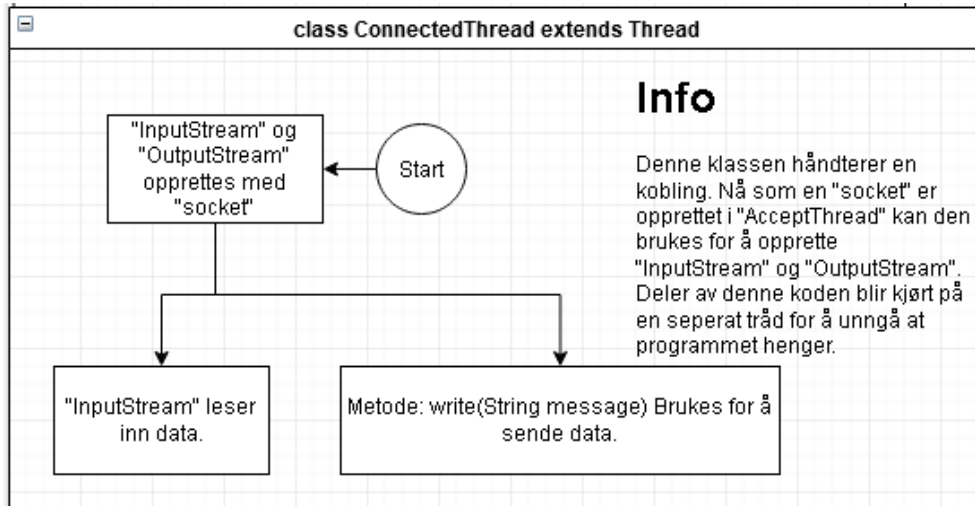
Serverapplikasjon

Kortleseren inneholder serverapplikasjonen. Kortleseren sin oppgave er å ta imot innkommende forespørsler om tilkobling fra klienter. Når en klient skanner en NFC-brikke mottar serveren IDen til brikken. IDen blir dermed sjekket opp mot databasen for evaluering. Hvis det er en gyldig bruker åpnes releet slik at porten er klar for gjennomgang. I dette kapitlet skal blåtann delen av applikasjonen forklares og illustreres. Det anbefales å følge med på (Figur 18, Figur 19, og Figur 20) mens gjennomgang foregår.

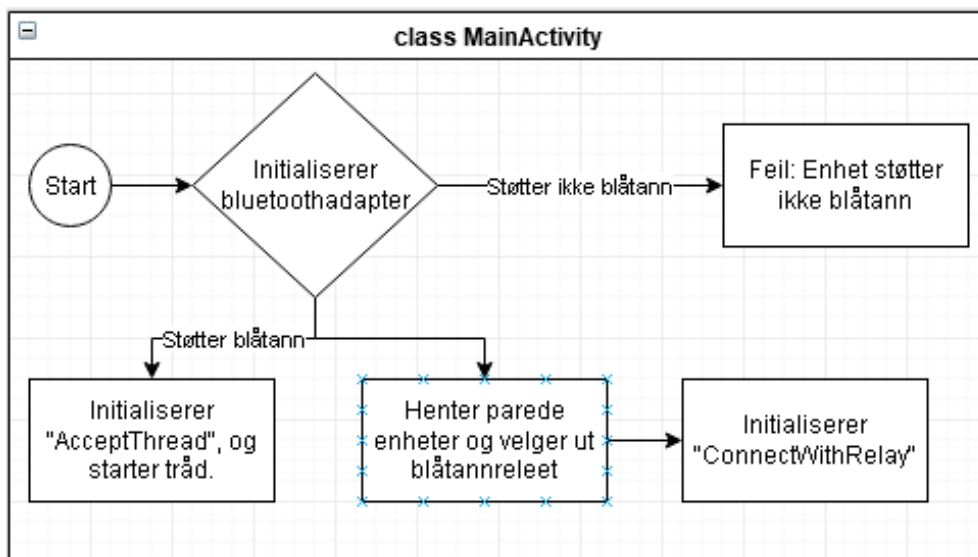
Kortleseren oppfører seg som sagt som en blåtannserver, men den fungerer også som en klient. Grunnen til dette er at releet som ble testet ikke støttet blåtannkonfigurasjon som betyr at enheten er låst til å oppføre seg som server. Dette var problematisk siden gruppen ønsket at kortleseren skulle ha rollen som server. Løsningen ble å sette opp kortleseren som server og klient. Grunnen til at vi må ha server på kortleseren er for å gjøre det mulig for klienter å koble seg til. Det er nemlig ikke mulig for to klientapplikasjoner å koble sammen. Server delen av applikasjonen er konfigurert til å akseptere innkommende forespørsler fra klienter med riktig UUID, mens klient siden av appen kobler seg til releet. Dette førte til betydeligere mer arbeid, men førte også til en mer fleksibel applikasjon.

Når serverapplikasjonen startes, vil applikasjonen lytte etter innkommende forespørsler fra klienter. Denne operasjonen foregår i klassen «AcceptThread» (Figur 20). Denne klassen bruker et «serverSocket»-objekt for å lytte etter klienter som forsøker å koble seg til serveren. Hvis en kobling blir akseptert knyttes tilkoblingen til en «socket» slik at koblingen kan håndteres. Koblingen blir håndtert i klassen «ConnectedThread» (Figur 18). Denne klassen bruker instansen av «socket» som ble opprettet i «AcceptThread» for å sende og motta data. For å sende data opprettes en «OutputStream». For innlesing av data brukes «InputStream». Klassene «AcceptThread» og «ConnectedThread» må kjøres på separate tråder slik at programmet ikke henger. Hvis disse klassene ikke hadde kjørt på separate tråder vil brukeren av applikasjonen oppleve at den fryser. Dermed kan det virke som at applikasjonen har låst seg, selv om applikasjonen opererer som den skal.

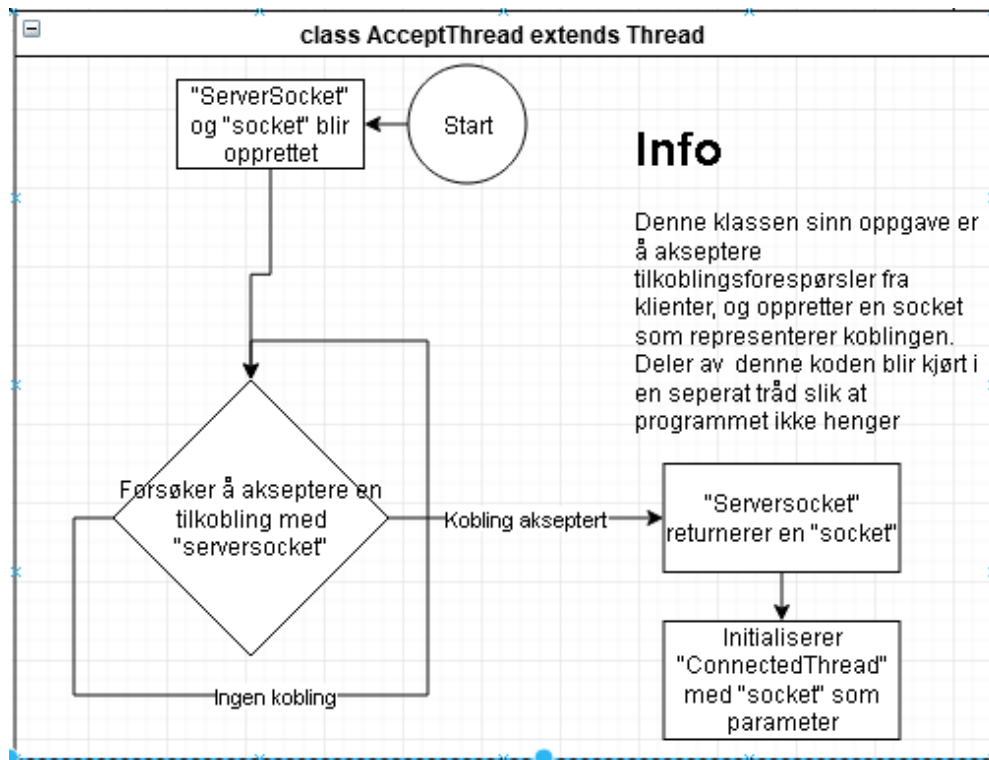
Samtidig som serverdelen kjører, vil applikasjonen koble seg til releet som klient. Koden som kreves for å opprette kobling til releet er veldig lik klientapplikasjonen som er forklart tidligere. Koden er utformet noe annerledes, men funksjonaliteten og resultatet blir det samme.



Figur 18: Flytskjema av "ConnectedThread" klassen



Figur 19: Flytskjema over "MainActivity" klassen



Figur 20: Flytskjema av "AcceptThread" klassen

3.7. Database

I dette prosjektet brukes database til register for om arbeider er innenfor byggeplass eller utenfor byggeplass. Arbeider kan få statusen endret ved innsjekking/utsjekking i adgangskontrollen til byggeplass. I tillegg kan en administrator holde oversikt og behandle databasen med en egen mobil applikasjon utviklet i dette prosjektet.

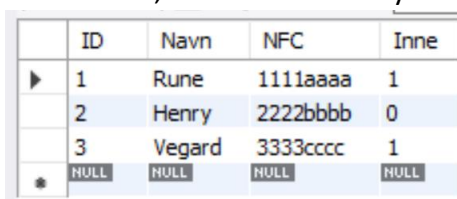
Det ble valgt å bruke MySQL som databasesystem i prosjektet med standard SQL spørrespråk som grensesnitt mot databaseserver. MySQL Workbench 8.0 brukes til oppsett og editering av databaseserver. I prosjektet ble det brukt en standard Windows 10 PC som database server. MySQL Workbench er noe arbeidsgruppen tidligere har fått opplæring via HVL fra emnet «Industriell IKT».

3.7.1. Hva er MySQL

MySQL er den mest populære Open Source SQL databaseadministrasjonssystemet. Det er utviklet, distribuert og driftet av Oracle [30]. SQL-delen av MySQL står for Structured Query Language, som betyr strukturert spørrespråk. En database er en strukturert samling av data, og MySQL er spesifikt en relasjonsdatabase. Dette vil si at informasjon er organisert i rader i tabeller, data er lagret som attributter og disse tabellene har relasjoner [31]. For å kunne bruke MySQL må du ha en server og en klient. Serveren er enheten som lagrer og styrer databasen. Klienten kobler seg til serveren via TCP/IP ved å legitimere seg med brukernavn og passord. Denne klienten sender en spørring til serveren og får en respons tilbake i form av en tabell. En spørring kan også legge til data i databasen. MySQL kan lagre og behandle ganske store datamengder, er fleksibelt, raskt og sikkert. [30]

3.7.2. Databasestruktur

Navnet på databasen er «anleggsterminal». Databasen inneholder tabellen «arbeidere» med kolonnene: ID, Navn, NFC, og Inne. Figur 21 viser tabellen «arbeidere» med eksempeldata. Forklaring tabell: ID primærnøkkelen i tabellen, og er heltalldatatype (INT), databasen ble programmert slik at server automatisk tilordner en unik ID når en ny bruker blir registrert (AUTO_INCREMENT). I kolonnen «Navn» lagres navnene til arbeiderene med datatype CHAR (30) (karakterer med maks 30 tegn). I kolonnen «NFC» lagres NFC-id til NFC-brikke som består av 8 siffer heksasmale tall, i databasen lagres dette som VARCHAR(8) (karakterer med maks 8 tegn). Kolonnen «Inne» har boolsk verdi «true» betyr at arbeider er innenfor arbeidsområde, mens «false» betyr at arbeider er ikke er innenfor arbeidsområde.



	ID	Navn	NFC	Inne
▶	1	Rune	1111aaaa	1
	2	Henry	2222bbbb	0
	3	Vegard	3333cccc	1
*	NULL	NULL	NULL	NULL

Figur 21: Tabellen «arbeidere» med eksempeldata

```

1 #Oppretter databasen "anleggsterminal"
2 CREATE DATABASE anleggsterminal;
3
4 #Bruker anleggsterminal som standard database
5 use anleggsterminal;
6
7 #Oppretter tabelegg med kolonner
8 create table Arbeidere(
9 ID INT NOT NULL AUTO_INCREMENT,
10 Navn char(30) NOT NULL,
11 NFC varchar(8),
12 Inne boolean DEFAULT false,
13 PRIMARY KEY (ID)
14 );

```

Figur 22: SQL skript for å opprette og utforme databasen i prosjektet

3.7.3. Databaseserver

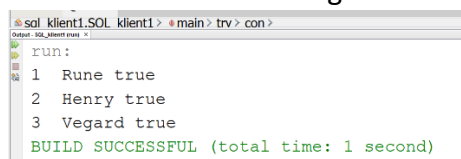
Å sette opp database med MySQL slik som i prosjektet krever veldig lite kode når en først har installert fungerende MySQL Workbench. Det å programmere databasen gjøres med spørrespråket SQL. Siden databasen bare har 4 kolonner ble kodingen av dette noen få linjer (se Figur 22).

3.7.4. Administratorapplikasjon

Å utvikle administratorapplikasjon for databasen til Android var mye mere omfattende enn å sette opp databaseserver. Utviklingen ble gjort over flere steg. Skjermbilder fra applikasjonen vises i **Feil! Fant ikke referanse-kilden.**

Steg 1: prototype i NetBeans

I dette steget var meningen å lage en javaapplikasjon som kunne hente og vise data fra tabellen «Arbeidere». Siden Java brukes som programmeringsspråk for Android applikasjoner og gruppen allerede er kjent med programmering av Java i Netbeans, men lite erfaring med Android studio, ble det først prøvd å lage en prototype applikasjon som kunne fungere i NetBeans. Til utveksling med database ble det brukt MySQL connector/j som er den offisielle driveren fra MySQL til kommunikasjon med server. Som hjelp til kodingen av en enkel klient ble det funnet og brukt noen veiledninger på nettet. Etter å ha prøvd å kode en prototype ble det likevel en del problemer og feilmeldinger fra driveren. Problemene løste seg etter hvert som med litt feilsøking med utgangspunkt i feilmeldingene der noe av problemet løste seg installere driveren på nytt med et annet versjonsnummer. Utskrift av tabellen i NetBeans se Figur 23.



```

sql klient1.SOL klient1 > * main > trv > con >
Output: SQL client true x
run:
1 Rune true
2 Henry true
3 Vegard true
BUILD SUCCESSFUL (total time: 1 second)

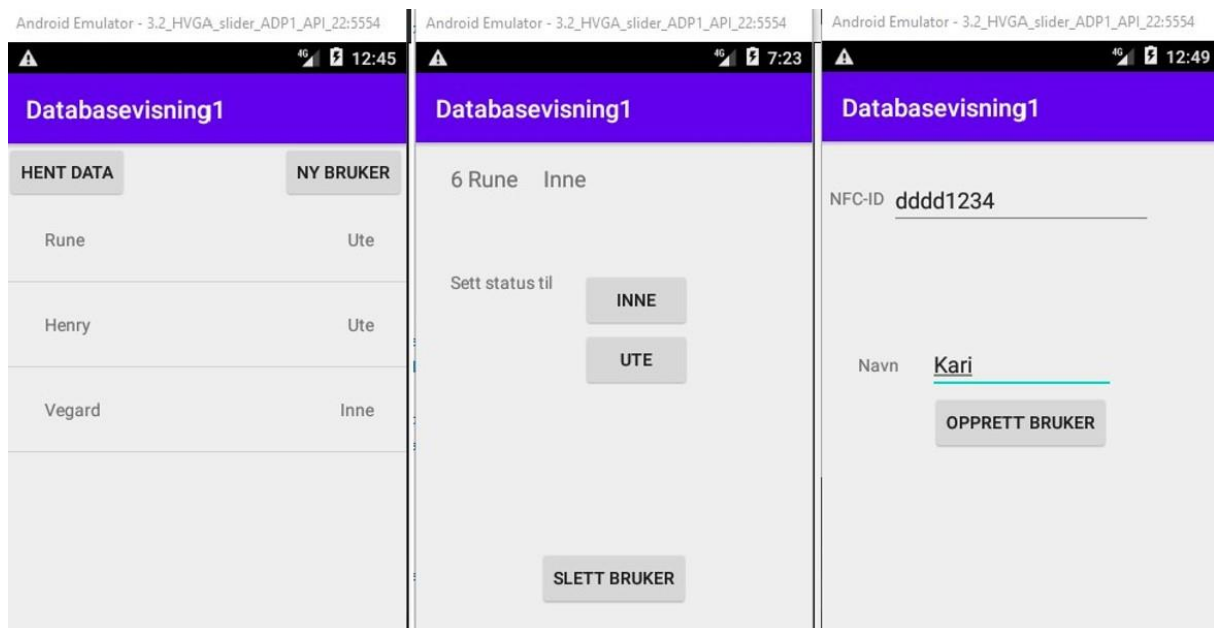
```

Figur 23: Utskrift av tabell «arbeidere» med klient i NetBeans

Steg 2: Listevisning i Android (hovedskjerm)

I dette steget var målet en overgang fra NetBeans til Android Studio. Det skulle lages en enkel app som skulle vise liste over arbeidere på et mobilt display og statusen til arbeiderene. Det ble funnet og brukt en veiledning på Youtube, denne viste hvordan en kunne lage en enkel Android applikasjon som kunne hente data fra tabell og vise det som en liste på skjermen [32]. Tanken var at om denne guiden fungerte så kunne det modifieres litt slik at den kunne passe til tabellen i prosjektet.

Etter at applikasjonen ble kodet var det mye som ikke fungerte fra starten, blant annet ville ikke den nyeste versjonen av SQL-driveren fungere fordi programmet er kodet for en eldre versjon av Android. Etter noe feilsøking fungerte systemet mere slik det skulle. Applikasjonen fra veiledningen ble modifisert til å vise arbeidere og statuser slik at det passet til prosjektet. Det ble programmert en «HENT DATA»-knapp over listen som en trykker på for å hente data. Imens listen oppdateres vil det vises statutstekst, der vil eventuelle feilmeldinger fra SQL vises. Senere ble det lagt til funksjon for automatisk henting av data hver gang hovedskjermen ble vist (for eksempel ved oppstart eller gå tilbake til hovedskjerm).



Figur 24: hovedskjerm, status, og ny bruker fra administratorapplikasjon

Steg 3: Skrive til database (endring av status)

Hittil hadde applikasjonen bare ett skjermbilde som viser liste over arbeiderene. I dette steget ønsket vi å utvide applikasjonen med funksjonalitet slik at en skulle endre om arbeider er utenfor eller innenfor byggeplass.

Dette ble gjort med å gjøre den eksisterende listen om til en klikkliste, samt lage et nytt skjermbilde som vises etter «trykk» på en linje i listen, der linjeposisjonen blir lagret som et valg. På skjermbildet ble det laget to knapper som brukeren kan trykke på for å sette status til inne eller ute for den valgte arbeider. Det ble også programmert inn funksjon for å slette bruker fra databasen med knappen «SLETT BRUKER», det ble også lagt inn ekstra sikkerhet med dialogboks med advarsel og valg mellom «Avbryt», eller «Ja» der bruker kan bekrefte eller avbryte slettingen. I bakgrunnen går det da en prosess som kontakter databasen og sender kommando til databasen. Ved eventuell feil, vil det vise feilmelding på skjermen.

Steg 4: Registrering av brukere og NFC.

Etter hvert så ble det bestemt at ID til NFC-brikke skulle legges til i databasen, samtidig som at en del av oppgaven var å lage funksjon for å registrere nye brukere. Når person sjekker inn på byggeplass med kortet sitt måtte IDen lest fra kortet sjekkes opp mot databasen for å deretter endre statusen til arbeider. Databasen ble derfor utvidet med kolonnen «NFC». På hovedskjermen ble det lagt til knappen «NY BRUKER», samt et nytt skjermbilde for inntasting av navn og NFC-ID, og med knapp «OPPRETT BRUKER».

3.8. NFC

3.8.1. Hva er NFC?

NFC er et akronym som står for «Near Field Communication» og er en trådløs teknologistandard for å overføre små mengder data mellom to enheter. Teknologien er basert på radiofrekvensfeltet og basefrekvensen er 13,56MHz [33], og bruker mindre enn 15 milliampere for kommunisere over avstander under 10 cm [34].

Spesifikasjonene for NFC er laget av NFC Forum, som er et konsortium av 170 selskap og har medlemmer som for eksempel Mastercard, Apple, Google og Nordic Semiconductors ASA [35]. For NFC er det en nødvendighet at minst en enhet er en transmitter, altså en aktiv enhet. Den andre enheten kan være passiv. Et eksempel på en aktiv enhet kan være en smarttelefon og et eksempel på en passiv enhet kan være en NFC-brikke som er i bankkortet ditt for kontaktløs betaling. Passive NFC-brikker har ikke en egen strømforsyning, kan ikke prosessere data og kan ikke koble seg til andre passive enheter. Aktive enheter kan både sende og motta/prosessere data. To aktive enheter kan kommunisere med hverandre og passive enheter [36]. NFC-brikker kan inneholde mellom 96 bytes og 8 kilobytes med data avhengig av brikketype, og overføringshastighetene er maksimalt 424 kilobits per sekund [36].



Figur 25: Logo for NFC [37]

3.8.2. Data i NFC

Data i NFC-brikker følger en standard som heter NFC Data Exchange Format (NDEF). Alle NDEF-meldinger består av to grunnleggende komponenter, en header og en «payload», heretter kalt header og datapakken. Headeren kommer først. Denne inneholder informasjon om overføringen. Informasjonen i headeren er størrelse, type, språk og IDen til datapakken. Headeren vil være mellom 5 og 12 bytes avhengig av hvilke felt som lagt inn. Typefeltet i headeren vil fortelle hva slags datatype som er i datapakken, som f.eks. tekst, URI, Smartposter. Datapakken inneholder selve dataen som skal overføres. Dataen kan være formatert på mange måter, både standardiserte og ikke-standardiserte. De fire vanligste typene er NDEF type 1 til 4. Det er pålagt for androidenheter med NFC-funksjonalitet å kunne lese disse. [38]

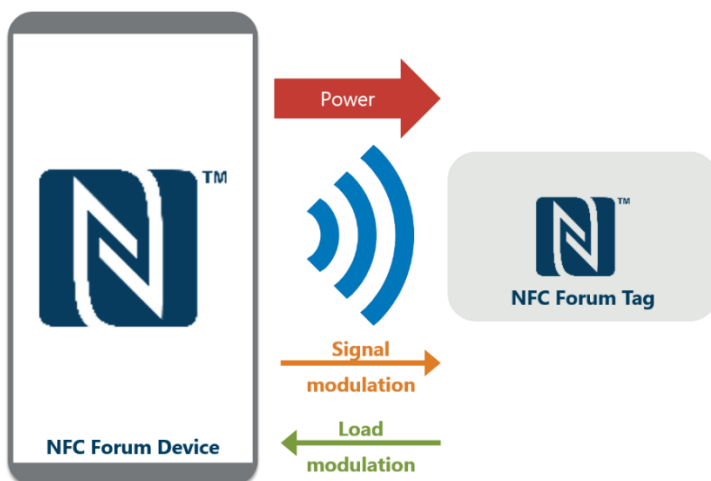
3.8.3. Sikkerhet

Som med all elektronikk er sikkerhet viktig. Spesielt siden NFC også blir brukt av diverse betalingstjenester. Det er hovedsakelig fire måter NFC kan utsettes for angrep. Disse fire er som følger: Avlytting, datakorrupsjon, datamodifikasjon og et såkalt man-in-the-middle-angrep. Men det kan være vanskelig å utføre siden NFC virker over så korte avstander. Den beste måten å sikre seg mot slike angrep er å opprette en sikker kanal mellom enhetene. En protokoll som kan brukes er Diffe-Hellmann. Diffe-Hellmann er en metode for utveksling av kryptografिनøkler, og vil gjøre kommunikasjonen mellom enheter relativt sikker. [39]

3.8.4. Hvordan virker NFC

NFC virker via magnetisk induksjon. En leser (aktiv enhet) sender ut en liten elektrisk strøm som skaper et magnetisk felt som skaper en bro mellom enhetene. Dette feltet blir mottatt av en lignende spole i den andre enheten (aktiv eller passiv).

Mottakerenheten gjør feltet om til elektriske impulser for å sende data, som identifikasjonsnummer, status og annen data tilbake til den aktive enheten. Passive enheter bruker energien fra det magnetiske feltet til sende responsen, mens aktive enheter har sin egen strømkilde og sender svaret via egne elektromagnetiske felt. Den aktive enheten sender informasjon til den passive enheten, for eksempel når den skal legge til informasjon i en NFC-brikke, ved å modulere radiofrekvensfeltet. Dette blir kalt signalmodulasjon. Når den aktive enheten skal lese data fra en NFC-brikke gjør den dette ved å måle modulasjon av lasten generert av brikken, altså lastmodulering. [33]



Figur 26: Illustrasjon av aktiv og passiv NFC-enhet [33]

3.8.5. NFC i vår app

Det ble først utviklet en selvstendig NFC-applikasjon før deler av denne ble implementert i klientapplikasjonen (blåtann). Den selvstendige NFC-applikasjonen kunne lese de fleste typer NFC-brikker og datatyper.

Klientapplikasjonen kan bare lese kort som er formatert etter NDEF-standarden og henter bare ut IDen til kortet. Dette er gjort fordi kortene som blir brukt for utviklingen av applikasjonen bruker dette formatet og IDen er tilstrekkelig for kunne unikt identifisere en bruker.

Det første som skjer når applikasjonen blir startet er at den sjekker om enheten som blir brukt har NFC-teknologi og om NFC er aktivert. Er ikke NFC-aktivert vil den åpne innstillingene for dette og be brukeren aktivere NFC. Når NFC er aktivert vil den aktivt lytte etter NFC-brikker. Applikasjonen gjør dette via intent. Intent er en klasse i Android og er en abstrakt beskrivelse av operasjon som skal utføres. Den kan blant annet bli brukt til å sette i gang aktiviteter. I dette tilfelle blir det opprettet et intentobjekt via intentklassen. Fra dette objektet blir IDen hentet ut ved hjelp av metoden `getByteArrayExtra` og argumentet `NfcAdapter.EXTRA_ID`. `getByteArrayExtra` og `NfcAdapter.EXTRA_ID` er metoder som er en del av Android. IDen blir lagret i en bytearray, som igjen blir omgjort til en hexadesimalstring via metoden `toHex`.

Etter at IDen er hentet ut blir den sendt videre til serverapplikasjonen via Blåtann og derfra videre til databasen via TCP/IP.

3.9. Sammenfletting av blåtann, NFC og database

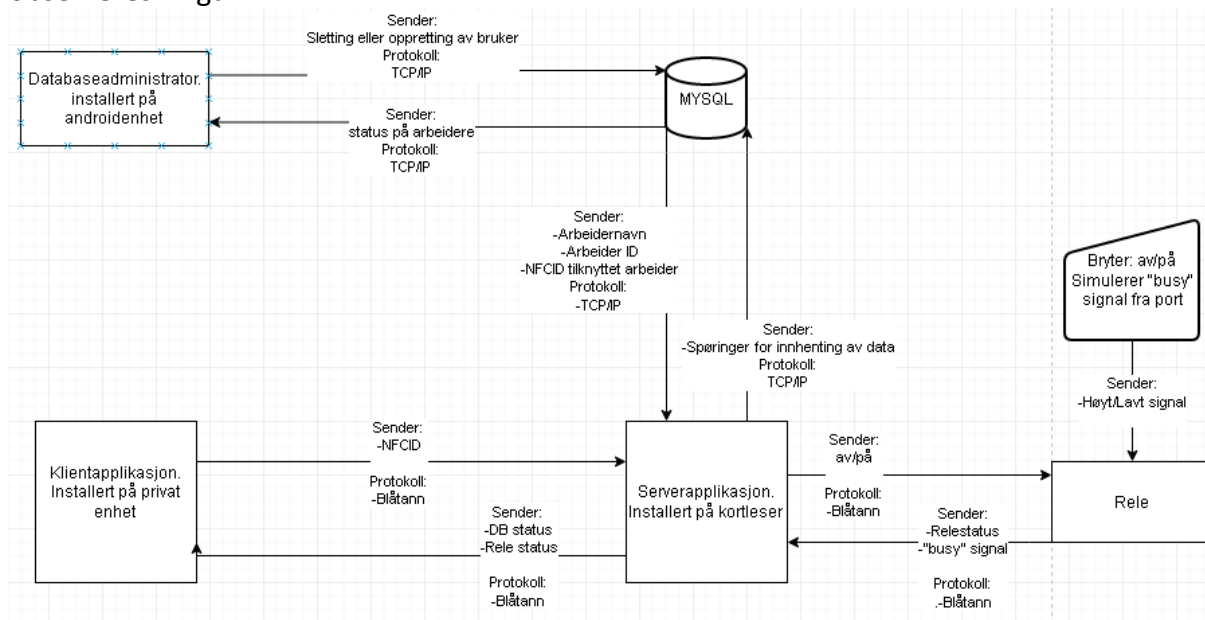
Blåtann, NFC og database applikasjonene ble først utviklet hver for seg. Det siste som gjensto var å legge til NFC i klientapplikasjonen, og gjøre kortleseren i stand til å kommunisere med databasen. NFC-biten i klientapplikasjonen ble implementert slik at en bruker kan skanne en brikke for å åpne porten. Å legge til denne funksjonaliteten gikk ganske greit siden testing allerede var utført i en separat NFC-applikasjon. I kortleseren ble det implementert kode som gjør det mulig å lese og skrive data fra databasen. Dette gikk også relativt smertefritt siden koden allerede var utviklet og testet i databaseadministratorapplikasjonen.

Når NFC og databasekommunikasjon var inkludert i applikasjonene ble det utført mye testing. I starten var det noen utfordringer med å få systemet til å fungere siden koden var av betydelig størrelse, og det var mange funksjonaliteter som skulle samarbeide.

Etter mye testing og feilsøking var den første prototypen av sluttproduktet klart, og det eneste som gjensto var finpussing, mer testing og noen små endringer.

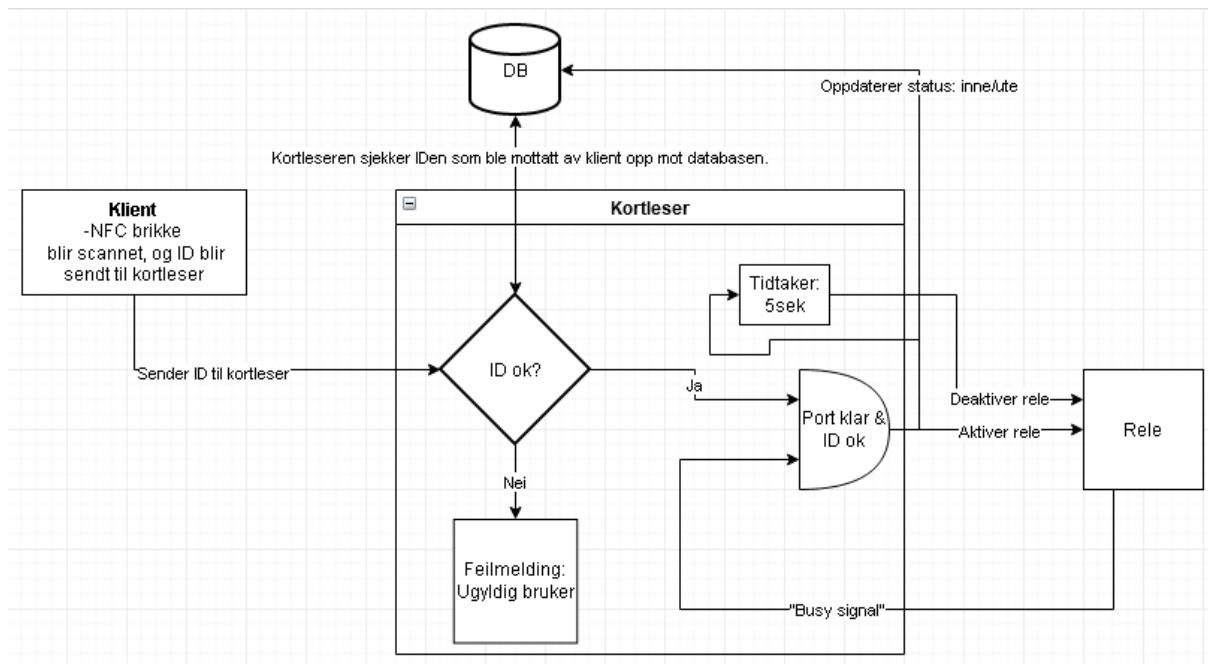
3.10. Sluttprodukt og resultat

Sluttproduktet består av tre applikasjoner, en database, og et rele som er innkapslet. Applikasjonene består av: klientapplikasjon (privat mobil), serverapplikasjon (kortleser som er montert ved port), og databaseadministratorapplikasjon. En oversikt over systemet kan observeres i Figur 27.



Figur 27: Flytskjema over hele systemet

Kortleseren som er montert ved rondellporten har i oppgave å åpne porten for personell som skanner en gyldig NFC-brikke på sin private mobil. Den skal også oppdatere statusen til arbeidere som går igjennom porten. Et flytskjema over hva som skjer når en klient skanner en NFC-brikke kan observeres i Figur 28.



Figur 28: Flytskjema som viser hva som skjer når en NFC brikke blir scannet

Når en arbeider har koblet sinn private mobil opp mot kortleseren, kan personen skanne en NFC-brikke for å åpne porten. Når en NFC-brikke blir lest av klientapplikasjonen vil den sende IDen til kortleseren. Kortleseren vil dermed sjekke om IDen eksisterer i databasen. Hvis IDen ikke eksisterer vil det komme opp en feilmelding som sier at brukeren ikke er registrert. Hvis IDen eksisterer, vil kortleseren åpne porten og oppdatere statusen til arbeideren. Hvis porten er opptatt, vil porten gi et «opptattsignal» til releet. Hvis porten er opptatt, vil brukere bli hindret i å logge seg inn eller ut.

Databaseadministratorapplikasjonen brukes til å administrere databasen. Applikasjonen viser en liste over arbeiderne som er registrert, og viser statusen til brukerne. Applikasjonen kan også brukes til å legge til eller slette brukere.

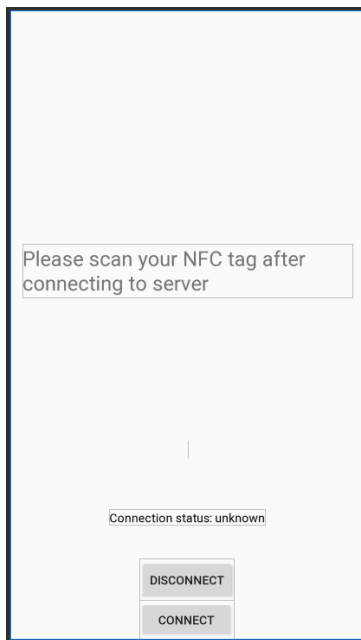
Selve releet er montert i en kapsling og har en trykkbryter som er koblet til GPIO. Denne bryteren simulerer «opptattsignal» fra rondellporten.

Bruk av serverapplikasjon

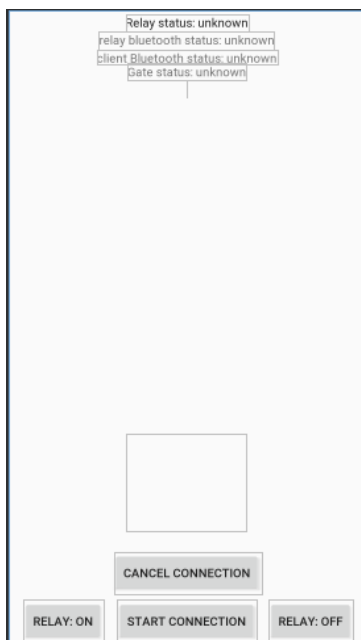
I Figur 30 kan GUIen til serverapplikasjonen(kortleseren) observeres. Hvis «start connection» trykkes vil applikasjonen koble seg til blåtannreleet, og lytte etter klienter som forsøker å koble seg til. Det er også en knapp for å avslutte koblingen: «cancel connection». Serverapplikasjonen støtter ikke NFC slik som klientapplikasjonen, men det er lagt inn knapper for å styre releet(relay: on, relay: off). Applikasjonen viser også informasjon som: «relestatus», «rele blåtannstatus», «portstatus», og «klient blåtannstatus». For å koble fra releet og klienten trykkes «cancel connection».

Bruk av klientapplikasjon

I Figur 29 vises GUIen til klientapplikasjonen. For å koble klienten til kortleseren trykkes «connect». Et infovindu(Connection status:) indikerer om tilkoblingen var vellykket. Når klienten har opprettet kontakt med kortleseren kan brukeren scanne en NFC-brikke for å åpne porten. Hvis brukeren ikke er gyldig, eller databasen er nede vil en feilmelding dukke opp. Det vil også være umulig å åpne porten hvis den er opptatt.



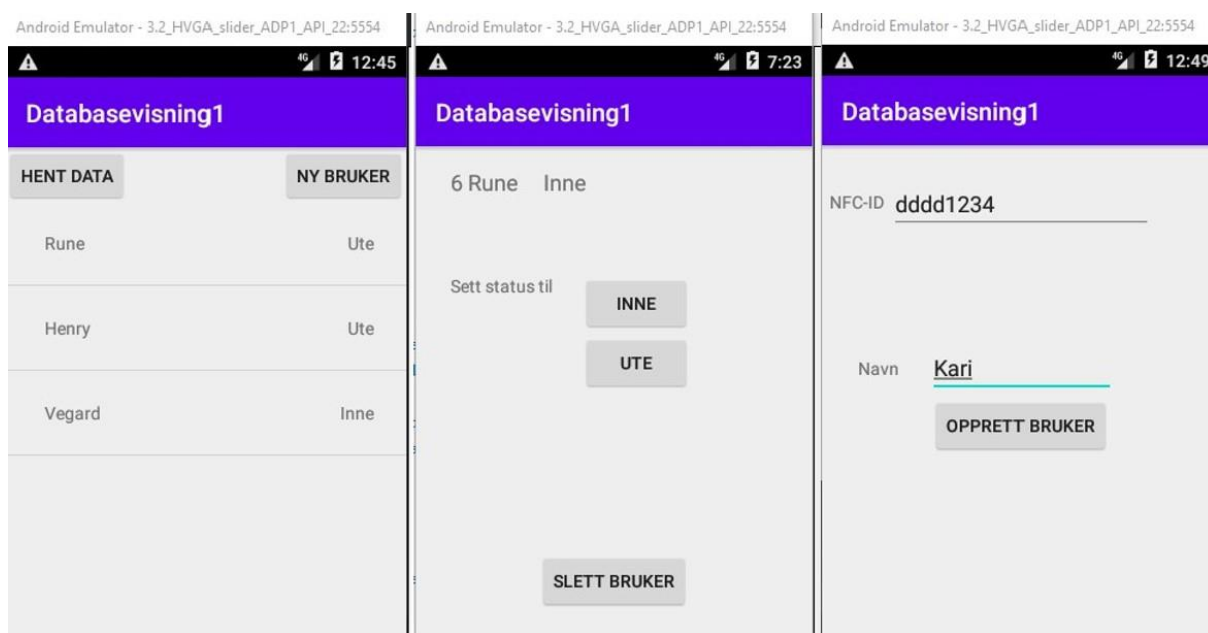
Figur 29: GUI til klientapplikasjon



Figur 30: GUI til serverapplikasjon

Bruk av administratorapplikasjon

Når administratorapplikasjonen kjøres, får bruker opp en hovedskjerm. Hvis knappen «hent data» trykkes vil applikasjonen hente arbeiderene, og deres status ifra databasen. Hvis brukeren ønsker å registrere ny arbeider kan «ny bruker» trykkes som vil gi en skjerm for ny bruker. I den nye skjermen blir brukeren spurt om å fylle inn navn og NFC-ID i de tomme feltene. Når feltene er fylt ut trykkes «opprett bruker». For å slette en bruker må brukeren av applikasjonen trykke på en spesifikk arbeider ifra listen i hovedskjermen. Da vil brukeren komme inn på en ny skjerm hvor det er mulig å slette arbeideren, eller oppdatere statusen. Figur 31 viser skjermbilder fra databaseadministratorapplikasjonen.



Figur 31: hovedskjerm, status, og ny bruker for serverapplikasjon. Duplikat av Figur 24

3.11. Endringer og avvik

Ved forprosjektfasen fant bachelorgruppen ut at det ble nødvendig med ekstra arbeid ved siden av Proresult AS sin oppgave. Derfor ble en liste laget over eventuelle utvidelser av systemet laget og vurdert. Nedenfor er en liste over oppgaver som ble lagt bort, eller endret:

- Målet om å kjøpe et relekort ferdig montert i innkapsling ble endret til å kjøpe innkapsling og relekort separat. Dette fordi vi ikke kunne finne noen som tilbydde et slikt ferdig produkt.
- Testing av relekort på rotasjonsgrind ble avlyst på grunn av covid-19.
- Ringeklokke. Tanken var at en ringeklokke skulle bli montert ved rondellporten slik at besøkende kunne få kontakt med personell. Ringeklokken skulle ikke bare lage lyd, men den skulle sende beskjed over SMS til bas eller forman. Denne komponenten skulle implementeres til slutt hvis gruppen hadde ekstra tid. Mot slutten av prosjektet ble finpussing og testing av eksisterende funksjonalitet prioritert.
- Ansiktsgjenkjenning var noe gruppen ønsket å implementere som en metode for å dra igjennom rondellporten. I starten ble denne løsningen undersøkt mye, men gruppen konkluderte med at denne teknologien ble for omfattende. Skulle dette ha vært en eventuell oppgave måtte ansiktsgjenkjenning vært hovedfokuset i bacheloroppgaven.
- Informasjonsskjerm. En informasjonsskjerm skulle monteres ved rondellporten som skulle vise status til arbeidere (inne/ute). Databaseapplikasjonen som ble utviklet viser en liste over arbeiderne og deres status, og virker som en erstatning for informasjonsskjerm.

3.12. Videreutvikling etter prosjektet

Finpussing og forbedring av eksisterende funksjonalitet

Etter at en prototype av prosjektet var utviklet er det klart at det er potensial for å forbedre produktet. Siden gruppen hadde ingen tidligere erfaring med Android utvikling, NFC, eller Blåtann var det en stor utfordring å lage en fungerende prototype. De viktigste forbedringene av applikasjonen vil være å gjøre produktet mer brukervennlig, og utføre mer testing for å gjøre programmet skikkelig robust mot feil. For eksempel er det informasjon som adresser, og navn på enheter hardkodet i applikasjonene. Hvis produktet skulle videreutvikles, ville det vært i fokus å finne bedre løsninger som fører til en langt mer brukervennlig opplevelse. Her er en liste over forbedringspotensial:

1. Minimere hardkodete (navn, adresser, etc...)
2. Utvikle en bedre GUI som gjør applikasjonene mer brukervennlig.
3. Utføre mye testing for å oppdage problemer.
4. Tidsstempel og utregning av samlet arbeidstid for arbeidere
5. Flere kolonner i database som for eksempel fornavn, etternavn, adresse, arbeidssted osv.
6. Funksjon for at arbeider kan legge inn valgfri tekst i status i database.
7. Automatisk henting av data i tidsintervall eller push på hovedskjermen.
8. Skanne NFC-brikke i stedet for manuell inntasting ved registrering av ny bruker i administratorapplikasjon.
9. Innstillinger som URL, Brukernavn, og passord fra app i stedet for hardkoding (administratorapplikasjon).

Utvidelser

I forprosjektrapporten ble det nevnt mange muligheter for utvidelse av prosjektet. Gruppen ønsket å kunne implementere:

1. Ansiktsgjenkjenning som innloggingsmulighet.
2. Ringeklokke ved rondellport for besøkende.
3. Informasjonsskjerm ved rondellport som viser hvem som er inne og ute.
(Databaseapplikasjonen kan brukes til dette)
4. Mulighet for flere klienter å koble seg til kortleseren samtidig
5. Automatisk tilkobling av klienter som er innenfor rekkevidde.
6. Automatisk frakopling av klienter som har gått gjennom port

3.13. Konklusjon

Målet med dette prosjektet var å styre en rotasjonsgrind med en androidapplikasjon. Dette ble gjort ved bruk av et blåttannrele som styrer låsemekanismen til porten. Ekstraoppgavene inkluderte en database som gjorde det mulig å ha en oversikt over arbeidere som er inne og ut av området. NFC ble også inkludert, slik at brukere kan skanne en NFC-brikke for å dra igjennom porten. Brukere har også mulighet til å installere en klientapplikasjon på deres private mobil. Denne applikasjonen gjør det mulig å skanne en NFC-brikke for å dra igjennom porten.

Arbeidsgruppen tror at kortet fra RelayPros er mere aktuelt for Proresult enn kortet fra Numato Lab, det på grunn av bedre muligheter for å konfigurere blåttann sikkerhet i kortet fra RealyPros. Dette relekortet dekker alle behovene for styring av rotasjonsgrind. Numato Lab sitt rele som ble brukt til testing fungerte fint for prototypen. Numato Lab sitt rele dekker alle behov når det gjelder funksjonalitet, men blåttannmodulen kunne ikke konfigureres. Siden blåttannmodulen ikke kan konfigureres er det ikke mulig å passordbeskytte releet. På grunn av dette er ikke Numato Lab sitt rele anbefalt av gruppen. Et annet godt alternativ er National Control Devices (NCD). NCD leverer relekort som kan tilpasses etter kundens behov. Releet vil dermed dekke alle behov som kreves av portstyringen. Den største fordelen ved å bruke releet som NCD tilbyr er at relekortet kan konfigureres til å bruke 24V driftsspenning. Dette betyr at relekortet kan kobles til strømforsyningen til rotasjonsgrinden. Muligens kan NCD også tilpasse relekort med innkapsling, og gjøre det klart til videresalg for Proresult. Bachelorgruppen anbefaler Proresult AS å vurdere NCD som en leverandør.

Innkapslingen som er brukt i prosjektet er lettmonterbar, har IP66, og har innebygget rekeklemme som gjør tilkobling enkelt. Innkapslingen passer godt til relekortet fra Numato, og vil trolig også passe godt til relekortet fra RelayPros.

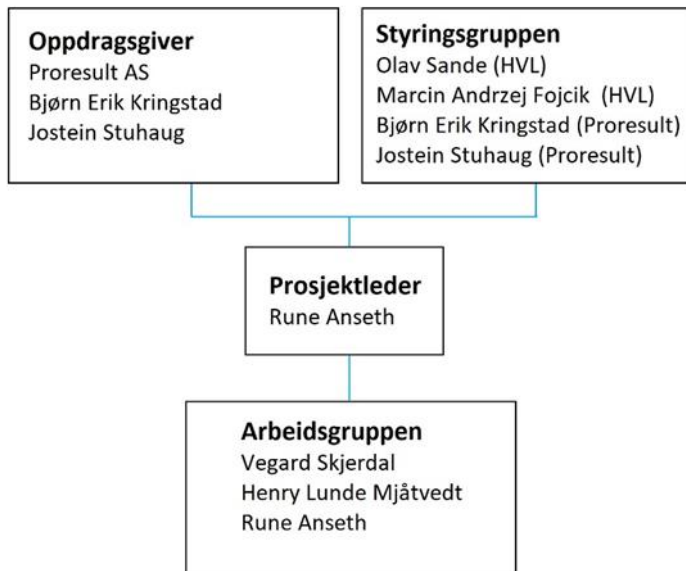
Programvaren som ble utviklet fungerer fint som prototype. Applikasjonene fungerer som planlagt, og styring av releet fungerer som det skal. Som sagt fungerer applikasjonene godt som prototype, og fungerer fint for testing av systemet. For at applikasjonene skal være klare for Proresult AS sitt produkt bør det utføres mye testing av programvaren. Bachelorgruppen fikk ikke mye tid til finpussing og testing av applikasjonene. Dette betyr at det mest sannsynlig ligger feil i programvaren som må oppdages og håndteres.

På grunn av Covid19 var det ikke mulig å teste systemet opp mot en rotasjonsgrind. Dette betyr at bachelorgruppen ikke kan garantere at løsningen fungerer med en rotasjonsgrind. Gruppen har gjort sitt beste å finne løsninger ved hjelp av dokumentasjonen til grinden. Den største utfordringen var å finne en spenningskilde til relekortet. Siden gruppen ikke fikk testet de ulike løsningene kan ikke bachelorgruppen anbefale en løsning som gir garantert resultat.

4. Prosjektadministrasjon

4.1. Organisering

Organiseringen bestod av tre nivå der oppdragsgiver og styringsgruppe er øverst, nedenfor er prosjektleder og nederst er arbeidsgruppen. Organisasjonskart, se Figur 32:



Figur 32: Organisasjonskart

Oppdragsgiver



Proresult AS ble stiftet i år 2007, de utvikler programvareløsninger til bygg- og anleggsbransjen. De har hatt sterk vekst som selskap. Det var først i 2019 at Proresult har hatt samarbeid med HVL. De tilbyr studentene gratis bruk av system til bruk for prosjekt, i tillegg til tilbud om bacheloroppgave. Proresult har kontor i Peak Sunnfjord i Førde. Kontaktpersoner fra Proresult: Bjørn Erik Kringstad (daglig leder), og Jostein Stuhaug (systemutvikler).

Styringsgruppen

Bestod av veilederne pluss kontaktperson fra oppdragsgiver. Hadde ansvaret for å ta prosjektets viktige avgjørelser som revidere og godkjenne prosjektmålene.

Prosjektleder

Arbeidsgruppen ble enig om at Rune Anseth er prosjektleder. Ansvaret til prosjektlederen var å administrere prosjektet. Dette inkluderte: Å kalle inn til møte, fordele oppgaver, dele opp prosjektet i passelig lange prosjektperioder.

Arbeidsgruppen

Bestod av studentene Rune Anseth, Vegard Skjerdal og Henry Lunde Mjåtvedt.

4.2. Prosjektarbeid og administrasjon

Oppstart prosjekt

Oppdragsgiver ble bestemt før jul. Arbeidsgruppen tok initiativ til å møtes og starte arbeidet tidlig i januar. Det første arbeidsgruppen gjorde var å ha et oppstartsmøte med Proresult. Deretter å lage et rammeverk for arbeidet, planlegge, og utforme en prosjektbeskrivelse.

Arbeidsmøter

Siden dette var et større skoleprosjekt så hadde gruppen møter ved behov og ofte (flere ganger i uken). Det viktigste her var å bli enige om hva som gjøres videre og fordele oppgaver som en skulle gjøres videre. Ofte ble det laget sakslister til møtene.

Styringsmøter

I prosjektperioden ble det innkalt til flere styringsmøter ved behov, disse møtene ble styringsgruppen og arbeidsgruppen med på. Første styringsmøte skjedde etter at forprosjektrapporten er levert. Intervall for styringsmøtene var omtrent hver 14.dag i hovedprosjektfasen, det ble laget møtereferat for hvert styringsmøte. Tema som ofte ble diskutert i styringsmøtene var: prosjektstatus, sluttrapport, oppgaver, endringer og avvik.

Webside

Det ble laget en webside for prosjektet. Denne inneholder noen korte meldinger om når milepæler er nådd, samt pressemelding og video. Rapporten er også lagt ut her.

<http://adgangskontroll.rf.gd>

Kommunikasjon med oppdragsgiver

I løpet av prosjektperioden hadde vi noen møter hos Proresult ved behov. Det ble ofte gjort skriftlig kommunikasjon via e-post og Microsoft Teams. Mesteparten av kommunikasjonen var det arbeidsgruppen som tok initiativ til. I kommunikasjon mellom parter var det viktig å unngå misforståelser, da misforståelser er en risikofaktor for prosjektet.

Logging

Til timeliste/arbeidslogg har vi brukt Proresult sitt system. Til å føre regnskap for utgifter er det brukt et exceldokument. Det ble også laget en møtelogg for møter der styringsgruppen var involvert.

Fordeling av oppgaver

Det ble ofte laget en oversikt over oppgaver som var å gjøre for prosjektet, oppgavene som kunne gjøres ble fordelt over arbeidsgruppen deretter.

Covid-19 innvirkning på prosjektet

Den 12 mars 2020 ble det innført mange sterke smitteverntiltak i hast fra myndighetene for å begrense spredningen av koronaviruset. Skolebygget ble stengt for studenter den 13.mars, alle skulle oppholde seg mest mulig hjemme. Gruppen fikk beskjed av veilederne at vi aller helst ikke skulle ha møter fysisk. Rune bor i en leilighet, mens Vegard og Henry bor sammen i et kollektiv, derfor ble det ikke mulig med flere møter samlet i same rom. Gruppen måtte ta høgde for at resten av skoleåret ble slik og alt skolearbeid måtte gjøres hjemmefra. Møtene skulle gjøres via videooverføringer. Presentasjonen for prosjektet skulle gjøres over video.

På grunn av disse smitteverntiltakene ble gruppen rammet på den måten at samarbeidet ble mindre tett, i tillegg til at noen psykiske faktorer kan ha påvirket prosjektarbeidet negativt. Eksempel på dette er at personer er i en annen «modus» hjemme enn på skolen.

Relekort ble testet hjemme, mens arbeidet med å teste komponenten på rotasjonsgrind ble avlyst, ellers skulle resten av oppgavene i prosjektet gjøres etter planlanen.

Mesteparten av oppgavene i prosjektet gikk greit å gjøre selvstendig. Noe av arbeidet er det best å gjøres sammen som for eksempel når slå sammen kode i programvaren, fullføre sluttrapport og å lage en presentasjon.

Verktøy for prosjektstyring

Vi har brukt Microsoft Onedrive for fildeling, en felles notatbok i Microsoft Onenote for diverse notater. I tillegg ble det brukt Team Gantt for å lage gantt-diagram. Vi har brukt Proresult sitt system til timeføring/arbeidslogg. Som en ekstra sikkerhet ble det tatt kopi av prosjektmappa til ekstern harddisk minst en gang i uken da tap av data ville gi fatale konsekvenser for prosjektet. Microsoft Teams ble brukt til noe av kommunikasjon med Proresult, i tillegg ble Microsoft Teams brukt til videomøter i perioden etter at skolebygget ble stengt.

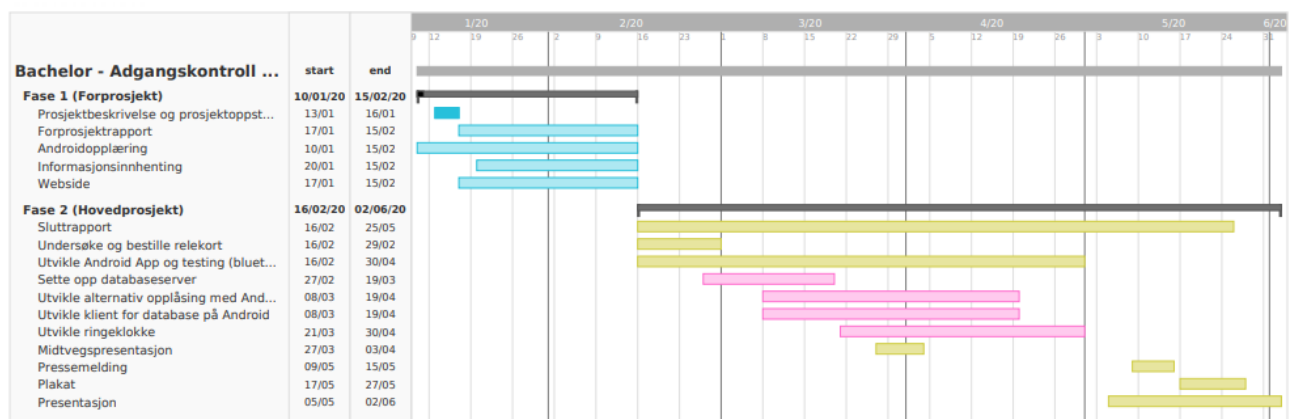
Utføring av praktiske oppgaver

Arbeidsoppgavene ble utført både gruppevis og selvstendig da begge arbeidsformer har forskjellige fordeler og ulemper. Vi har hatt litt mer selvstendig arbeid enn gruppearbeid. Når vi arbeidet gruppevis ble det det gjort i skolebygget frem til skolebygget ble stengt. Etter at skolebygget ble stengt ble det gjort samarbeid i videomøter.

4.3. Gjennomføring i forhold til planen

Fremdrift

Prosjektet ble periodisert i fasene forprosjekt og hovedprosjekt. Forprosjektperioden skulle være frem til forprosjektrapporten ble godkjent. Hovedprosjektet var resten av prosjektet. Den planlagte fremdriften var et reviderbart dokument. Figur 33 viser gantt diagram for fremdriften som var planlagt. De oppgavene med blå tidslinje er for forprosjektperioden, i hovedprosjekt er de med gul linje de oppgavene som er høy prioritet, mens de med rosa linje hadde lavere prioritet.



Figur 33: Gantt diagram planlagt fremdrift

Tabell 2: Tidsskjema

Oppgave	Planlagt ferdig	Faktisk ferdig
Prosjektbeskrivelse	16.01	16.01
Forprosjektrapport	15.02	21.02
Webside	15.02	21.02
Bestille relekort	29.02	09.03
Utvikle Android App og testing	30.04	22.05
Databaseserver	19.03	23.03
Programmering	30.04	22.05
Pressemelding	15.05	18.05
Plakat	27.05	Etter levering av sluttrapport
Presentasjon (arbeide med)	02.06	Etter levering av sluttrapport

Kommentar til gjennomføringen

Den overfladiske oppgaveplanen, se Tabell 2, viser hvordan prosjektet er utført etter tidspunkt planlagt. Vi har ikke vært veldig strenge på å følge den planlagte tidsskjemaet, har fulgt planen med noen avvik.

4.4. Økonomi og tidsbruk

Regnskap

Nedenfor er tabell som viser prosjektets faktiske kostnader, og budsjetterte kostnader (se Tabell 3).

Tabell 3: Regnskap og budsjett

Beskrivelse	Rekneskap (NOK)	Budsjettert (NOK)
Bluetooth modul til prototype	280	280
Relekort	3292	2500
Leie av rondellport (30 dager)	0	7000
NFC-brikker	0	150
SUM	3572	9930

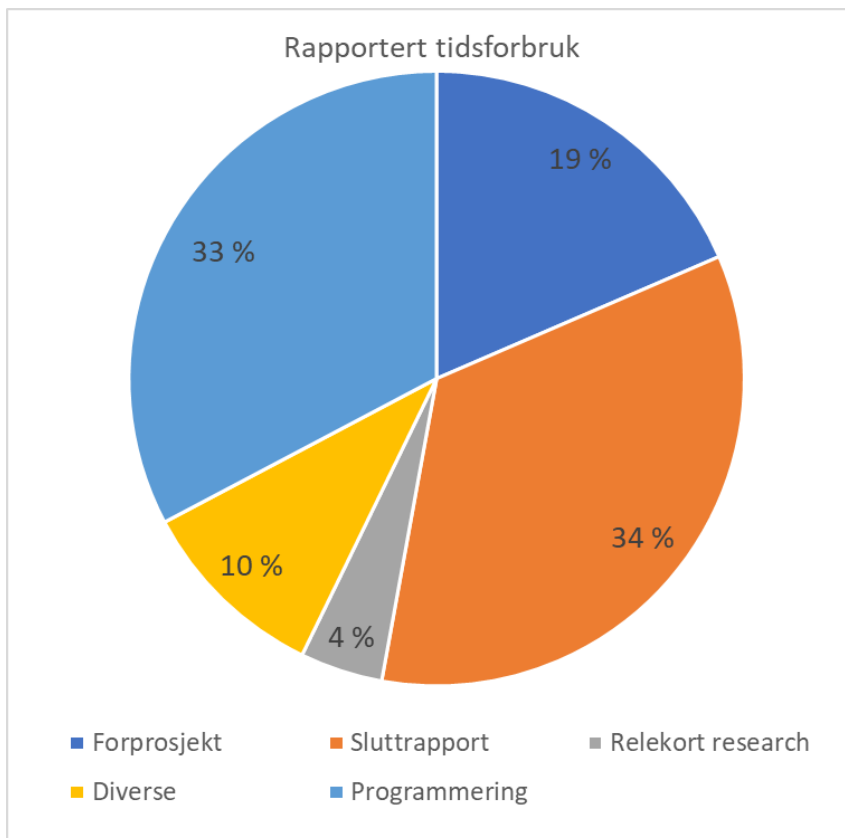
Kommentar til budsjett og regnskap: På grunn av at det ikke noe testing av fysisk rotasjonsgrind, derfor ble det heller ingen kostnader på rotasjonsgrind. I budsjettet ble det planlagt å kjøpe inn et relekort, men det ble kjøpt to forskjellige derfor ble det høyere kostnad på relekort. NFC brikker for testing ble planlagt innkjøpt, men arbeidsgruppen fant ut at studentkortene kunne brukes som alternativ. Budsjettet ble laget i forprosjektet.

Arbeidstimer

I bacheloroppgaven er den forventede arbeidsmengden ca. 500t/student, derfor ble totale arbeidstiden for prosjektet planlagt til 1500 timer. Tabell 4 viser planlagte og forbrukte arbeidstimer for gruppe medlemmene. Forbrukt arbeidstid er hva hver arbeider har rapportert. Sektordiagrammet (Figur 34) viser rapportert total arbeidstid kategorisert.

Tabell 4: Arbeidstimer planlagt og forbrukt

Gruppedlem	Planlagt arbeidstid (timer)	Forbrukt arbeidstid (timer)
Rune	500	434,6
Vegard	500	386,1
Henry	500	408,7
Totalt	1500	1229,4



Figur 34: Sektordiagram rapporterte arbeidstimer

4.5. Generell prosjektevaluering

Måloppnåing

For oppdragsgiver sin del ønsket de seg et produkt som skulle være så ferdig som mulig. I prosjektet har vi fått utviklet en løsning det kreves videreutvikling med før det kan selges som et produkt til kunder. Vi fikk utviklet en androidapplikasjon som kan styre et blåttannrele, på den måten er hovedmålet oppnådd.

I resultatmålet er det definert at det skulle utvikle et «robust produkt». For å kunne bekrefte om produktet er robust eller ikke, så kreves det grundig testing av systemet og helst på en reell rotasjonsgrind. Det ble ikke mulig å gjøre andre tester enn hjemmetesting.

Når det gjelder utvidelsesmålene er de fleste mål oppnådd, unntaket er ringeklokke og ansiktsgjenkjenning. Ansiktsgjenkjenning ble forsøkt arbeidet med, men viste seg at det ble for komplisert. Ringeklokke var det målet som hadde lavest prioritet og ble aldri påbegynt. Arbeidsgruppen er fornøyd med måloppnåingen blant utvidelsesmålene.

Erfaringer med prosjektgjennomføringa

Igjennom dette prosjektet har vi erfart om samarbeid og kommunikasjon er en betydelig del av arbeidet i denne type prosjekt. I starten av prosjekt er det viktig å være nøye med å forstå oppgaven fra oppdragsgiver for å unngå misforståelser. Møter bør være godt planlagt og strukturert om de skal være effektive. Videomøter er som oftest et godt alternativ til tradisjonelle møter, mens fysisk oppmøte fungerer best i praktiske oppgaver. For administrasjon er det viktig å sørge for god oppgavefordeling arbeidsgruppen og passe på konstant fremdrift, oppdatere hverandre ofte om statuser, og avtale nye arbeidsoppgaver så fort noen har lite å gjøre. Skriftlige gjøremålslist er et godt hjelpemiddel da det ellers fort kan bli glemt. Vi har også erfart er at det er lett å henge seg opp i og gjøre mye arbeid på enkelte detaljer som kanskje ikke er så viktig for prosjektresultatet, og samtidig kan en kan glemme hva som er viktig for helhetsresultatet.

Risikoanalyse er en del av prosjektet, men det er ikke lett å tenke ut alle typer risikofaktorer. Koronaviruset hørte vi om i forprosjektperioden, men ingen tenkte det var en risikofaktor for prosjektet før skolebygget ble stengt. Gruppen tror konsekvensene Covid-19 kan ha hatt for prosjektresultatet er at kvaliteten på arbeidet har blitt noe redusert.

Når det gjelder programmering er det noen ganger enklere enn forventet og andre ganger vanskeligere enn forventet å lære seg noe nytt. Det å lære seg androidutvikling var litt mere krevende å lære seg enn ventet, men etter hvert så mestret vi det godt selv om vi lærte det på egenhånd med hjelp av fritt tilgjengelig informasjon på nett. Andre erfaringer vi har gjort er at når en skal lage et salgbart produkt er det viktig å undersøke forskrifter som gjelder, for eksempel CE-merking.

Referanser

- [1] «FX910,» Famoco , [Internett]. Available: <https://www.famoco.com/fr/produits/pointeuse-badgeuse-intelligente/fx910/>. [Funnet 19 mai 2020].
- [2] «Bluetooth Relay 1-Channel 5-Amp ProXR Lite,» RelayPros, LLC, [Internett]. Available: https://relaypros.com/Relay/Device/R15PL_ZBT. [Funnet 19 mai 2020].
- [3] «2 Channel Bluetooth Relay Module,» Numato Systems, LLC, [Internett]. Available: <https://numato.com/product/2-channel-bluetooth-relay-module>. [Funnet 22 mai 2020].
- [4] «Relæ-print med 2 stk. skifte-relæer. Styres via USB eller BT/Wifi ad-on moduler,» Dansk Binær teknik A/S, [Internett]. Available: <https://danbit.no/no/traadloes/14415-info-kommer.html>. [Funnet 22 mai 2020].
- [5] «BT1 - Blåtann Leser / Rele / Kortleser med APP (Mifare),» Holars AS, [Internett]. Available: <https://holars.no/produkt/12002/bt1-blatann-leser-rele-kortleser-med-app-mifare/>. [Funnet 22 mai 2020].
- [6] Ahlsell, «Ahlsell - Koblingsboks DK 0606 G,» [Internett]. Available: <https://www.ahlsell.no/33/elektro/gr.-11---25-installasjonsmateriell/12---ror-tak--og-veggbokser-koblingsmateriell/bokser-fittings/boks--kapsling-for-innfelt-montering-i-vegtak/1270304/>. [Funnet 15 04 2020].
- [7] «Turkey's first and leading turnstile producer...,» Özak Geçiş Teknolojileri San. Tic. A.Ş, [Internett]. Available: <https://ozak-t.com/Corporate/About-Us>. [Funnet 19 mai 2020].
- [8] «BTX 300 N1 S / D,» Özak Geçiş Teknolojileri San. Tic. A.Ş, [Internett]. Available: <https://ozak-t.com/Products/Pedestrian-Access-Control-Series/Full-Height-Turnstiles/BTX-300-N1-S-D>. [Funnet 19 mai 2020].
- [9] «BTX 300 N1 TECHNICAL SPESIFICATIONS,» [Internett]. Available: https://ozak-t.com/upload/Dosyalar/Urunler/Yaya-Gecis-Sistemleri/Boy-Tipi-Turnikeler/BTX300S-D-N1/Dosyalar/TD-1302-0085_3%20BTX%20300%20N1%20TECHNICAL%20SPECIFICATIONS.pdf. [Funnet 19 mai 2020].
- [10] «CE-merking,» Standard Norge, [Internett]. Available: <https://www.standard.no/standardisering/ce-merking/>. [Funnet 15 mai 2020].

- [11] «MERKE CE,» Forbrukerrådet, [Internett]. Available: <https://www.forbrukerradet.no/merkeoversikten/andre/ce/>. [Funnet 20 mai 2020].
- [12] «Ny metode - EU-direktiv, forordning og norsk tilsynsorgan,» Standard Norge, [Internett]. Available: <https://www.standard.no/standardisering/ce-merking/tilsynsorgan/>.
- [13] «CE-merking,» Nemko, [Internett]. Available: <https://www.nemko.com/no/sertifisering/produktsertifisering/sertifisering-europa/ce-merking/>. [Funnet 20 mai 2020].
- [14] «FX100,» Famoco SAS, [Internett]. Available: <https://www.famoco.com/android-devices/handheld-devices/fx100/>. [Funnet 17 mai 2020].
- [15] «EDUCBA,» [Internett]. Available: <https://www.educba.com/java-vs-kotlin/>. [Funnet 19 Februar 2020].
- [16] «Kotlin,» [Internett]. Available: <https://kotlinlang.org/docs/reference/comparison-to-java.html>. [Funnet 19 Februar 2019].
- [17] C. Hoffman, «How to Pair a Bluetooth Device to Your Computer, Tablet, or Phone,» 5 juni 2017. [Internett]. Available: <https://www.howtogeek.com/214185/beginner-geek-how-to-pair-devices-over-bluetooth/>. [Funnet 17 April 2020].
- [18] W. Stallings, «Piconets and Scatternets | Introduction to Bluetooth | InformIT,» 26 Oktober 2001. [Internett]. Available: <https://www.informit.com/articles/article.aspx?p=23760&seqNum=4>. [Funnet 12 Mai 2020].
- [19] M. J. C. M. Z. J. C. Farhat Saleemi, «Routing technique for inter slave bonding in Bluetooth scatternets formation,» 5 Februar 2010. [Internett]. Available: https://www.researchgate.net/publication/228990705_Routing_technique_for_inter_slave_bonding_in_Bluetooth_scatternets_formation. [Funnet 14 April 2020].
- [20] Honeywell, «Bluetooth Secure Simple pairing - FileField,» Honeywell, 24 November 2014. [Internett]. Available: https://support.honeywellaidc.com/servlet/fileField?entityId=ka02K000000DIQyQAK&field=File_1__Body__s. [Funnet 20 April 2020].
- [21] schemobotics, «HC-05 6 pin Wireless Bluetooth Rf Transceiver Module Serial or TTL or RS232 for Arduino,» schemobotics, [Internett]. Available: <http://schemobotics.com/product/hc-05-6-pin-wireless-bluetooth-rf-transceiver-module-serial-or-ttl-or-rs232-for-arduino/>.

- [22] Android, «Bluetooth Overview | Android Developers,» Android, 19 Februar 2020. [Internett]. Available: <https://developer.android.com/guide/topics/connectivity/bluetooth#ConnectDevices>. [Funnet 12 Mars 2020].
- [23] Android, «BluetoothAdapter | Android Developers,» Android, 5 Mai 2020. [Internett]. Available: <https://developer.android.com/reference/android/bluetooth/BluetoothAdapter>. [Funnet 3 Februar 2020].
- [24] Android, «BluetoothDevice | Android Developers,» Android, 5 Mai 2020. [Internett]. Available: <https://developer.android.com/reference/android/bluetooth/BluetoothDevice>. [Funnet 5 Februar 2020].
- [25] Android, «BluetoothSocket | Android Developers,» Android, 5 Mai 2020. [Internett]. Available: <https://developer.android.com/reference/android/bluetooth/BluetoothSocket>. [Funnet 5 Februar 2020].
- [26] Android, «InputStream | Android Developers,» Android, 13 Februar 2020. [Internett]. Available: <https://developer.android.com/reference/java/io/InputStream>. [Funnet 7 Februar 2020].
- [27] Android, «OutputStream | Android Developers,» Android, 13 Februar 2020. [Internett]. Available: <https://developer.android.com/reference/java/io/OutputStream>. [Funnet 15 Februar 2020].
- [28] Android, «UUID | Android Developers,» Android, 27 Desember 2019. [Internett]. Available: <https://developer.android.com/reference/java/util/UUID>. [Funnet 17 Februar 2020].
- [29] Android, «BluetoothServerSocket | Android Developers,» Android, 27 Desember 2019. [Internett]. Available: <https://developer.android.com/reference/android/bluetooth/BluetoothServerSocket>. [Funnet 15 Februar 2020].
- [30] MySQL, «MySQL 8.0 Reference Manual - What is MySQL,» [Internett]. Available: <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>. [Funnet 20 05 2020].
- [31] Store Norske Leksikon, «relasjonsdatabase - Store Norske Leksikon,» [Internett]. Available: <https://snl.no/relasjonsdatabase>. [Funnet 20 05 2020].

- [32] «Android Studio For Beginners Part 4,» Google LLC, [Internett]. Available: <https://www.youtube.com/watch?v=bu5Y3uZ6LLM>. [Funnet 19 mai 2020].
- [33] NFC Forum, «NFC Forum - About the technology,» [Internett]. Available: <https://nfc-forum.org/what-is-nfc/about-the-technology/>. [Funnet 1 Mai 2020].
- [34] J. Sathyan, N. Anoop, N. Narayan og S. V. Kizhakke, i *A Comprehensive Guide to Enterprise Mobility*, CRC Press, 2012, p. 375.
- [35] NFC Forum, «NFC Forum - Our members,» [Internett]. Available: <https://nfc-forum.org/about-us/our-members/>. [Funnet 30 04 2020].
- [36] R. Triggs, «What is NFC and how does it work,» Android Authority, [Internett]. Available: <https://www.androidauthority.com/what-is-nfc-270730/>. [Funnet 10 05 2020].
- [37] NFC Forum, «NFC Forum Logo asset guidelines,» [Internett]. Available: <https://nfc-forum.org/nfcforum-logo-assets-guidelines/>. [Funnet 15 05 2020].
- [38] Android, «Android Developer - Ndef,» [Internett]. Available: <https://developer.android.com/reference/android/nfc/tech/Ndef>. [Funnet 05 05 2020].
- [39] Electronic Notes, «NFC Security - Electronic Notes,» [Internett]. Available: <https://www.electronics-notes.com/articles/connectivity/nfc-near-field-communication/security.php>. [Funnet 15 05 2020].
- [40] «RND 320-00038 - Pluggbar strømforsyning, 12V, 1.5A, 18W, RND Power,» Distrelec Group Inc, [Internett]. Available: <https://www.elfadistrelec.no/no/pluggbar-stromforsyning-12v-5a-18w-rnd-power-rnd-320-00038/p/30098158>. [Funnet 19 mai 2020].
- [41] «L7812CV - Linear Fixed Voltage Regulator, 12V, 1.5A, TO-220, ST,» Distrelec Group Inc, [Internett]. Available: https://www.elfadistrelec.no/no/linear-fixed-voltage-regulator-12v-5a-to-220-st-l7812cv/p/17309016?q=*&pos=1&origPos=2&origPageSize=10&track=true. [Funnet 19 mai 2020].

Vedlegg

1. Bakgrunnsinformasjon fra Proresult
2. Komponentliste
3. Møtelogg
4. Budsjett og regnskap
5. Programkoder
 - 5.1. Blåtannserverapplikasjon (Android studio-prosjekt)
 - 5.2. Blåtannklientapplikasjon (Android studio-prosjekt)
 - 5.3. Databaseadministratorapplikasjon (Android studio-prosjekt)
 - 5.4. Kildekoder (HTML-format)
 - 5.4.1. Administratorapplikasjon
 - 5.4.2. Klientapplikasjon
 - 5.4.3. Serverapplikasjon
 - 5.5. Databaseskript (MySQL)
 - 5.6. Bruksanvisning
6. Dokumentasjoner
 - 6.1. Bruksanvisning Numato Lab relekort (link)
 - 6.2. Dokumentasjon BTX-300 S
 - 6.2.1. Installasjonsguide BTX 300 S
 - 6.2.2. Spesifikasjoner BTX 300 S
 - 6.3. Produktark Famoco FX100
 - 6.4. Dokumentasjon Hensel DK 0606 G
 - 6.5. Dokumentasjoner RelayPros (syv dokument)
7. Videodemonstrasjon (link)
8. Arbeidstimer

Vedlegga er lagt i zip-fil. Filer og mapper er nummerert.



Høgskulen på Vestlandet

Bacheloroppgåve

H02-300

Predefinert informasjon

Startdato:	14-05-2020 15:00	Termin:	2020 VÅR
Sluttdato:	25-05-2020 14:00	Vurderingsform:	Norsk 6-trinns skala (A-F)
Eksamensform:	Bacheloroppgåve		
SIS-kode:	203 H02-300 1 O 2020 VÅR		
Intern sensor:	Marcin Fojcik		
Intern sensor:	Olau Sande		

Deltaker

Naun:	Rune Anseth
Kandidatnr.:	308
HVL-id:	160265@hvl.no

Informasjon fra deltaker

Antall ord *:	15034	Inneholder besvarelsen konfidentielt materiale?:	Nei	Jeg bekrefter at jeg har registrert oppgavetittelen på norsk og engelsk i StudentWeb og vet at denne vil stå på vitnemålet mitt *:	Ja
Egenerklæring *:	Ja				

Gruppe

Gruppenavn:	Adgangskontroll
Gruppenummer:	8
Andre medlemmer i gruppen:	Vegard Skjerdal, Henry Lunde Mjåtvedt

Jeg godkjenner avtalen om publisering av bacheloroppgaven min *

Ja

Er bacheloroppgaven skrevet som del av et større forskningsprosjekt ved HVL? *

Nei

Er bacheloroppgaven skrevet ved bedrift/virksomhet i næringsliv eller offentlig sektor? *

Ja, Proresult AS