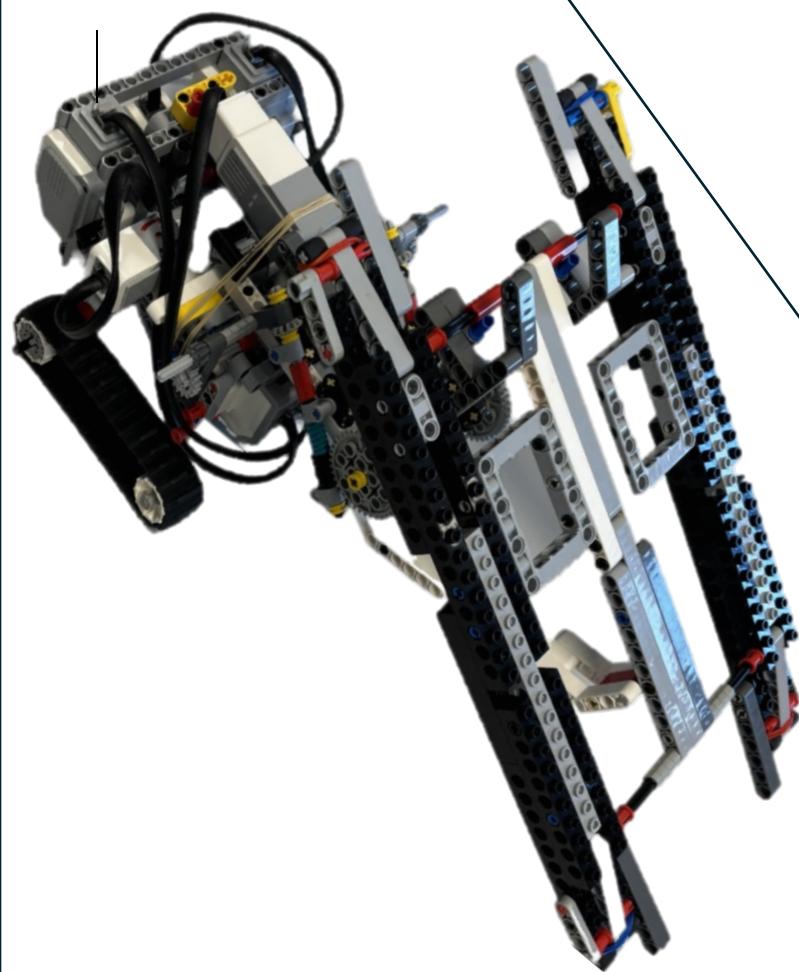


**2024**

**IDATT1004**

**Teambasert  
samhandling**



# Brobygger - SpanX

Prosjektoppgave av Team 15

Erlend Sundsdal, Stian Closs Walmann, Tord Johannessen Fosse,  
Christian Petter Remman

## Forord

Hensikten med prosjektet har vært å jobbe tett som et felles utviklingsteam med samme prosjekt og mål.

Vi tok utgangspunkt i denne problemstillingen: «Hvordan kan et team designe og bygge en samfunnsnyttig og bærekraftig Lego robot, som kan bidra til å løse utfordringer knyttet til infrastruktur i krevende områder?». Med denne problemstillingen konkluderte teamet at de skulle lage en brobyggerrobot. Videre, arbeidsmetoden som brukes i faget vil med stor sannsynlighet være relevant for alle medlemmer av teamet i arbeidslivet. Derav er det å tilegne seg kunnskap om å jobbe i team et mål. Dette innebærer blant annet å skrive timelister, ukerapporter, møteinkalling og møtereferat, og å sette opp en arbeidskontrakt.

Gjennom prosjektet har teamet lært om prosjekt som arbeidsform, både om etablering, organisering, gjennomføring og rapportering. Samt identifisere, planlegge og gjennomføre et lite utviklingsprosjekt i samarbeid med andre.

Gjennom idemyldring, og deretter klassifisering av ideer, kom teamet frem til at en brobyggerrobot er den oppgaven som best balanserer det å utnytte gruppens ferdigheter, bærekraft og samfunns-nytte, og hva gruppen har lyst til å utføre.

Takk til Shiza Ahmad for tilbakemeldinger og hjelp til å forbedre rapporten.

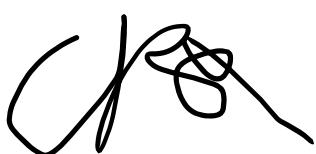
20.11.2024, Gløshaugen, Trondheim, Norge

Stian Closs Walmann

Tord Johannesen Fosse



Cristian Petter Remman



Erlend Sundsdal



## Sammendrag

Team 15 i Teambasert samhandling ved NTNU Trondheim har utviklet en innovativ brobyggerrobot som en del av avslutningsprosjekt. Denne roboten konstruert av LEGO Mindstorms, er designet for å plassere midlertidig, kritisk infrastruktur i form av broer i områder med manglende eller ødelagt infrastruktur.

Teamets prosjekt er basert på en kombinasjon av teammedlemmene individuelle interesser og lidenskaper, samtidig som det fokuserer på samfunnsbehov og adresserer FNs bærekraftsmål. Gjennom utviklingsprosessen har teamet lagt vekt på god kommunikasjon og dette med å støtte hverandre slik at teamet blir det mest effektive og flytende det kan være. Teamet har også fordelt arbeidet på en slik måte at alle i teamet kan vise sine gode sider innenfor hver av prosjektets hovedpilarer, dette værende:

- **Programmering** – Utvikling av algoritmer for robotens bevegelse og funksjonalitet.
- **Design** – Effektiv konstruksjon av robotens fysiske utseende, samt tilleggsdeler som broen, og alt mekanisk til roboten.
- **Prosjektleddelse** – Koordinering av teamets innsats og framdrift, samt bestemme neste prosjektmål, og delegere med tanke på disse.

Som sagt lå mye av bakgrunns-tanken til prosjektet i samfunnsnyttighet og FNs bærekraftsmål. Derfor vil roboten vår kunne spille en potensiell viktig rolle i:

- **Redningstransport** – Rask gjenoppretning av tilgang til utsatte områder etter naturkatastrofer
- **Militære operasjoner** – Tilrettelegging for rask mobilitet i utfordrende terrenge.
- **Sivil infrastruktur** – Midlertidige, alternative infrastrukturelle løsninger under vedlikehold eller reparasjon av broer.

Teamet har programmert roboten til å navigere utfordrende terrenge, vurdere optimale broplasseringer, og plassere midlertidige broer med minimal menneskelig inngrip. Gjennom flere tester og evalueringer har robotens fysiske design, samt programvare blitt betydelig oppgradert.

Videre, når det kommer til FNs bærekraftsmål adresser vår brobygger spesifikt disse:

**Mål 9** – Innovasjon og infrastruktur – ved å tilby innovative løsninger for infrastrukturelle problemer [1].

**Mål 11** – Bærekraftige byer og samfunn - ved å styrke samfunnsmotstandsdyktighet mot katastrofer [1].

Til slutt, har dette prosjektet gitt teamet gode erfaringer innenfor teamarbeid, samt styrket vårt individ sine egenskaper innenfor; prosjektleddelse, design, og programmering. Alle har fått et læringsrikt innblikk i hvordan et team går fra startfasen, hele veien til slutfasen med et produkt.

## Abstract

Team 15, in IDATT1004(Teambasert samhandling) at NTNU Trondheim, have developed an innovative bridge-builder robot as part of the end term assessment. This robot constructed from LEGO Mindstorms, is designed to place temporary, critical infrastructure in areas that either lack, have damaged, or destroyed infrastructure. These temporary bridges are highly modular and can be placed and picked up by the robot alone.

The project is based on a combination of each team member's individual interests and passions, while focusing on societal needs and addressing the UN's Sustainable Development Goals. Throughout the development process, there has been an emphasis on good communication between team members, and support of each other's responsibilities. This has made the team as effective and fluid as possible.

Furthermore, the workload has been distributed in a way that the group members can showcase their strengths within each of the project main pillars. These being:

- **Programming** – Development of algorithms for the robot's movement and functionality
- **Design** – Effective construction of the robot's physical appearance, as well as additional parts such as the modular bridge, and all mechanical aspects the bridge-builder contains
- **Project management** – Coordination of the team's efforts and progress, as well as determining the next project goals, and delegating with these in mind.

As mentioned earlier, many of the underlying ideas behind the project lay in societal utility and the United Nations Sustainable Development Goals. Therefore, the robot might play a potentially important role in:

- **Rescue transport** – Rapid restoration of access to affected areas where natural disasters have occurred.
- **Military operations and aid** – Facilitating rapid mobility in challenging terrain
- **Civil infrastructure** – Temporary, alternative infrastructure solutions during maintenance and repair of bridges

This robot is programmed to navigate challenging terrain, assess optimal bridge placements, and place temporary bridges with minimal human intervention. Through several tests and evaluations, the robot has improved both physical in design, and more accurate by updated software. Furthermore, when it comes to the UN's Sustainable Development Goals, our bridge-builder robot specifically addresses these goals:

**Goal 9 – Innovation and Infrastructure** – by offering innovative solutions for infrastructural problems [1].

**Goal 11 – Sustainable Cities and Communities** – by strengthening community resilience towards natural disasters [1].

Finally, this project has given each team member good experience in teamwork, as well as strengthening each individual's skills within project management, design, and programming. Everyone has gained an educational insight into how a team progresses from the initial start phase to the final phase, with a product in hand.

## Innholdsfortegnelse

Forord.....	2
Sammendrag .....	3
Abstract .....	4
Innholdsfortegnelse.....	6
Introduksjon .....	8
Teamet .....	9
Erlend Sundsdal.....	9
Stian Closs Walmann .....	10
Tord Johannessen Fosse.....	11
Christian Petter Remman .....	12
Problembeskrivelse.....	13
Resultater .....	13
Valg av oppgave.....	13
Utfordringer .....	14
Møter og planlegging.....	14
Arbeidsfordeling .....	15
Legobygging.....	15
Kodeprosess .....	15
Diskusjon .....	16
Arbeidsfordeling .....	16
Bruk av samhandlingsverktøy.....	18
Kommunikasjon.....	18
Målbevissthet .....	19
Planmessighet.....	19
Tidsbruk .....	19
Resultat.....	20
Konklusjon og anbefalinger.....	21
Bærekraft og samfunnspåvirkning.....	21
Bibliografi .....	23

Vedlegg til prosjektrapporten.....	25
Arbeidskontrakt (Vedlegg 1) .....	25
Prosjektplan (Vedlegg 2) .....	31
Møteinnkallinger og møtereferat (Vedlegg 3).....	35
Aktivitetsdiagram (Vedlegg 4).....	39
Skisse av utseendet til førsteutkast av roboten (Vedlegg 5).....	40
Timeliste med statusrapport (Vedlegg 6) .....	41
Presentasjon av prosjektet (Vedlegg 7) .....	48
Refleksjonsnotat (Vedlegg 8).....	52

## Introduksjon

Dette dokumentet presenterer utviklingen av brobyggerroboten SpanX, som er en del av avslutningsoppgaven i emnet IDATT1004 – Teambasert samhandling ved NTNU, Trondheim. SpanX er designet for å håndtere utfordringer knyttet til midlertidig infrastruktur, spesifikt plassering og opplukking av broer. Roboten som er bygget med Lego Mindstorms EV3, er utviklet for å legge broer raskt og effektivt, og kan spille en viktig rolle i nødhjelpsoperasjoner, militære oppdrag og andre kritiske situasjoner hvor etablering av rask og sikker infrastruktur er nødvendig.

Formålet med dokumentet er å gi en strukturert beskrivelse av prosjektets mål, gjennomføring og resultater. Det inkluderer en beskrevet og reflektert gjennomgang av teamets arbeid, fra idemyldring og valg av oppgave, til de utfordringene og løsningene som ble implementert.

Strukturen til rapporten er som vedlagt i innholdsfortegnelsen over, her er et kort overordnet utsnitt:

- *Forord, sammendrag, introduksjon*
- *Teamet, problembeskrivelse*
- *Resultater, diskusjon og refleksjon*
- *Konklusjon og anbefalinger*
- *Bærekraft og samfunnspåvirkning*
- *Vedlegg*

## Teamet

### Erlend Sundsdal



Hei! Jeg heter Erlend Sundsdal, er 21 år og kommer fra Kristiansand. Jeg går første året dataingeniør på NTNU Trondheim. Interesser jeg har er å lage mer eller mindre nyttige «oppfinnelser», terrengsykling og gitarspilling. Tidligere har jeg hatt to år på folkehøgskole i Alta, første år som elev på friluftsliv 70° nord linjen, og andre året som stipendiat på Arktis-Svalbard, med oppfølging av en botreningselev som hovedfokus.

Tidligere erfaring med teknologi og programmering før studier begrenser seg til hobbyprosjekter hjemme. Dette har jeg stor glede og noe erfaring av, men lite erfaring med å jobbe team.

**Stian Closs Walmann**

## Litt om meg

Hei! Jeg er Stian Walmann, 23 år og studerer dataingeniør ved NTNU Trondheim. Jeg har bred bakgrunn fra forsvaret hvor jeg har jobbet med både operativ og teknisk tjeneste. Dette har gitt meg gode inngangsverdier til prosjektarbeid og til å samarbeide med andre mennesker.

På sikt kan jeg se for meg å dra tilbake til forsvaret hvor jeg vil kunne bidra med ny innsikt og god læring fra studiet og arbeidslivet.

**Tord Johannesen Fosse**

## Litt om meg

Hei! Jeg heter Tord Fosse, er 23 år gammel og studerer dataingeniør ved NTNU i Trondheim. Gjennom livet har jeg alltid vært interessert i alt som har med datamaskiner å gjøre. Med fagbrev som dataelektroniker har jeg bygd opp erfaring innenfor service, brukerstøtte, feilsøking og vedlikehold av ulike systemer.

Jeg liker alt som har med vann å gjøre, alt fra bading og seiling til surfing og dykking. På fritiden er jeg også interessert i film og serier.

### Christian Petter Remman



## Litt om meg

Hei! Jeg heter Christian Petter Remman, er 20 år gammel og studerer dataingeniør ved NTNU Trondheim. Min oppvekst har gitt meg et unikt perspektiv, med røtter både i Norge og Australia. Jeg er en aktiv person som elsker idrett, spesielt bandy, golf, løping og styrketrening. Tidligere har jeg drevet mitt eget enkeltpersonforetak innen E-Commerce, noe som har gitt meg verdifull erfaring.

Jeg er ambisiøs, og mitt langsigtige mål er å utvikle noe fra bunnen av. Jeg vet ikke helt hva dette vil bli enda – kanskje en startup eller en programvareløsning – men jeg ønsker å skape noe stort med mine egne to hender.

## Problembeskrivelse

I startfasen av prosjektet tolket teamet oppgaven som å utvikle en prototype av en samfunnsnyttig robot ved hjelp av Lego Mindstorms, med fokus på FNs bærekraftsmål. Tolkningen ble at koden til prototype roboten skulle være objektorientert, og tenkte Python som programmeringsspråk på grunn av tidligere oppgaver i faget.

Videre startet teamet en omfattende brainstorming som bare fokuserte på kvantitet. Dette ledet til at teamet fikk en stor mengde med ideer, ca 35, som var av varierende kvalitet. Etter avstemning, diskutering og refleksjon rundt hver av ideene, så landet teamet på denne «brobyggerroboten» som ide. Den endelige problemstilling ble «Hvordan kan team 15 utvikle en mobil, fleksibel og effektiv brobyggerrobot prototype ved hjelp av Lego Mindstorms, som kan forbedre mobilitet og tilgjengelighet i områder med manglende, eller ikke eksisterende infrastruktur?».

Prosjektet går ut på å konstruere en prototype av denne brobyggerroboten, som raskt kan sette ned, samt plukke opp midlertidig broer over elver, kløfter eller andre hindringer. Roboten skal være mobil nok til å bevege seg i utfordrende terrenget, og bygge broer som tåler både mennesker og lettere kjøretøy. Målet er å bidra til infrastruktur, med effektive, midlertidige løsninger – som kan spille en kritisk rolle for både nødhjelp og militære operasjoner, spesielt i situasjoner etter naturkatastrofer hvor tradisjonelle broer kan være ødelagt.

Dette prosjektet er fokusert rundt FNs bærekraftsmål om; industri, innovasjon, og infrastruktur, bærekraftige byer og samfunn, og klimahandling. Ved å fokusere på katastrofehåndtering og forbedring av infrastruktur i utsatte områder, bidrar prosjektet til å bygge mer bærekraftig samfunn.

Vedlegg 5 er første utkast av utseende til roboten, rett etter teamet hadde bestemt seg for en ide, og grovt diskutert hvordan roboten skal fungere, og se ut.

## Resultater

### Valg av oppgave

Teamet valgte oppgaven basert på en blanding av idemyldring, diskusjon og eliminasjon. Utgangspunktet vår var å samle alle ideene på ett sted og stemme frem de ideene som var gjennomførbare, samfunnsnyttige, og interessante. Deretter trakk teamet frem forslagene med flest stemmer og diskuterte rundt hvilke som var mest aktuelle. Teamet kom frem til at en brobyggerrobot var ideen som var mest aktuell.

## Utfordringer

På programvarefronten opplevde teamet at Gitlab var vanskelig å utnytte på en fornuftig måte. Teamet merket i de første oppgavene at det ble veldig kunstig å bruke Gitlab til å administrere og koordinere dobbelt i og med at det var snakk om relativt små og raskt løsbare oppgaver. Her merket teamet i ettertid at prosjektet kunne blitt gjort litt mer effektivt av bedre og mer integrert bruk av Gitlab sine samarbeidsfunksjoner.

En av de andre utfordringene teamet møtte underveis i prosessen som gruppe var upålidelige sensorer som ga målinger som ikke stemte. Den skyldige sensoren var en infrarød sensor for avstandsmåling som både feilfunksjonerte og var feilplassert. Kombinasjonen av disse problemene førte til at feilsøkingsprosessen ble veldig vanskelig da gruppen ikke visste hva som var feil før teamet isolerte sensoren og gjorde tester på den alene. Det viste seg etter mye prøving og feiling at sensoren viste veldig lang avstand om den lå for nærmek bakken. Grunnen til at det tok mye tid å finne denne feilen er at disse målingene var ulogiske og ikke samsvarer med hva sensoren faktisk skal vise. Løsningen var da å ombygge roboten og flytte sensoren lengre fra bakken så målingene ble riktige. Dette førte også til at teamet måtte endre parameterne til avstandsmålingen. Dette ga en god peker på hva slags feil gruppen kan møte på, noe som bidro til problemløsingsevne hos teamet.

## Møter og planlegging

Før hver time hadde teamet et lite stand-up møte hvor koordinering av dagsplanen, og oppsummering av sist økts arbeid var sentralt. I tillegg har det vært veiledningsmøter med læringsassistent som har vurdert teamet på hvor godt teamarbeid og planprosess har blitt utført. Ved starten hadde teamet et lengre møte hvor gruppen la opp og bestemte hvordan tidsbruken på de forskjellige områdene skulle være. I etterkant av disse større møtene (med læringsassistent og oppstartsmøte) noterte teamet de identifiserte oppgavene i Gitlab og lagde issues i issueboardet. Hvert stand-up møte i de vanlige timene tok teamet til seg i issues i issueboard og planla i utgangspunktet hvordan disse skulle løses, men teamet tok også stilling til nylig identifiserte problemer som krevde umiddelbar handling. Generelt var møtene effektive, noe som gjorde at gruppen kastet bort minimalt med tid og fikk gjort mye arbeid.

I vedlegg 6 ligger oversikt over forbruk av tid på dokumentasjon, koding og robotutvikling.

## Arbeidsfordeling

Gjennomgående har arbeidsfordelingen fungert bra i teamet. Fra start av falt teamet inn i naturlige roller som alle holdt seg til. Dette har gitt gode resultater ettersom alle har vært komfortable med de valgte oppgavene og har synes det har vært relativt givende arbeid. Fordelingen under prosjektet har vært to på programmering, én på bygging og én på rapport. Dette har vært en god fordeling ettersom alle har vært pragmatiske, tilpasningsdyktige og har bidratt ved å legge inn ekstra innsats ved behov.

## Legobygging

Basen av broleggeren tok utgangspunkt i en vanlig bulldoser. Dette ble valgt fordi den har lavt tyngdepunkt, godt grep og god fremkommelighet. Alle disse kvalitetene ble sett på som nyttige i forhold til en brolegger som i teorien skal fungere i ulendt terren og steder der det er vanskelig å komme fram. I tillegg er lavt tyngdepunkt viktig for å holde balansen når broen skal legges ut da det blir mye vekt som flyttes rundt langt unna tyngdepunktet. Teamet plasserte motorene bakerst av samme grunn. Motorene fungerer som motvekt når broen legges ned.

Armen ble det gjort mange modifikasjoner på siden gruppen i utgangspunktet var begrenset til en motor. Teamet trengte en konstruksjon som kunne både løfte, holde for transport og legge ned bro med en bevegelse. Her ble det laget en armmekanisme med en ekstra arm som skal gli ut under broen for å støtte opp lenger ut enn der broen er festet. Dette har som hovedfunksjon å støtte opp så broen kan legges langt ut fra broleggeren. Uten ekstraarmen vil broen falle rett ned, noe som begrenser bredden på avgrunner og elver man kan krysse.

I drivverket til broleggingsfunksjonen har teamet lagt inn flere momenter som skal sørge for mer stabil, sikker og repeterbar brolegging. Det er blant annet mye gir-reduksjon som sørger for at armen og motoren er sterkt nok til å holde broen helt ytterst. Her har gruppen også lagt inn snekkeveksel som ikke kan drives baklengs av bro-armen. Dette er avgjørende for å kunne ha en så tung bro som mulig. Det er også mange brikker og strikker som er lagt inn for å sørge for at alle tannhjul sitter der de skal og ikke hakker når broen er lengst ute.

## Kodeprosess

I starten av kode-prosessen var ikke roboten ferdig sammensatt. Utenom dette kunne det fortsatt begynnes med oppsett av koden og lage klasser for de forskjellige funksjonene til roboten. Det var diskusjoner om bruken av Java som programmeringsspråk, ettersom at oppgaven spesifiserte objektorientert koding. Etter forslag fra andre grupper og studentassisterenter, ble det enighet om Python som kodespråk. Dette var i stor

del på grunn av god dokumentasjon og enkelt oppsett av LEGO Mindstorm med bruken av Python, selv om ingen av gruppemedlemmene hadde kodet objektorientert i Python før. [2]

Etter bestemt kodespråk, måtte selve kodingen begynne. Ettersom at roboten måtte kunne kjøre fremover, bakover og svinge i forskjellige grader, ble det laget en klasse for kjøring (Drive). Roboten måtte også ha muligheten til å løfte og sette ned broen. Det ble derfor også laget en klasse for dette kalt BuildBridge. For å kjøre selve programmet ble det laget en Main-klasse, og en Test klasse for testing av koden. Ved legging av broen, var det viktig at roboten pekte rett ut mot fallet der broen skulle legges. Det ble derfor laget en metode i Drive som skulle rette opp roboten før broen skulle legges. For at denne metoden lettest mulig skulle gjennomføres, ble det lagt til en gyrosensor på roboten. Ved å bruke gyrosensoren kan roboten stoppe i en vinkel som er gunstig før broleggingen.

Til å lagre og dele koden med resten av teamet ble GitLab. Her lastet gruppen opp ny kode ettersom den ble laget. For at opplastningen av koden ikke skulle ødelegge for noen andres sin kode, ble det laget nye branches, slik som Drive branch og BuildBridge branch. Ved å bruke disse branchene kunne koden også kvalitetssikres av andre gruppemedlemmer når koden skulle merges til main. For å teste hele koden måtte roboten være ferdig bygget, men ved å teste en og en metode kunne testingen starte tidlig. Etter at det dukket opp feilmeldinger underveis ble disse løst ved å samarbeide og se på dokumentasjonen av pybricks [2].

I slutten av kode-prosessen, etter at roboten var ferdig bygget, ble variabler som tid, rotasjon og hastighet justert for å lage en mest mulig stabil brolegger. Disse variablene kunne enkelt endres i mainfunksjonen, siden klassene tok inn disse variablene som verdier til bruk i funksjonene. Etter at koden var ferdigstilt, ble dokumentasjonen i koden justert for å gi alle på gruppen en god forståelse av hva som skjer til enhver tid.

## Diskusjon

Prosjektet i teambasert samhandling har vært en lærerik opplevelse, hvor teamet har identifisert styrker og svakheter ved arbeidet. Mange av utfordringene som ble møtt er relevante i forhold til arbeidsliv og generelt i møte med andre mennesker. Teamet har jevnt over vært fornøyd med både arbeidsfordeling og kommunikasjon internt i teamet, riktignok har gruppen bitt merke i følgende læringsmomenter:

## Arbeidsfordeling

I SpanX prosjektet begynte teamet med en relativt flytende arbeidsfordeling. Etter hvert som prosjektet utviklet seg, ble det tydelig hvem som hadde hovedansvar for de ulike delene, noe som bidro til en effektiv arbeidsprosess.

- **Erlend** og **Tord** hadde delt hovedansvar for programmeringen. De delte programmet i to deler og tok ansvar for hver sin. Erlend fikk da hovedansvar for programmering av broleggingsfunksjonaliteten. Det innebar å programmere hvordan broen legges og plukkes opp. I motsetning, hadde Tord hovedansvar for alt som handlet om kjøring og generell forflytting og bruk av sensorer for roboten. Denne fordelingen fungerte godt i utgangspunktet når koden grovt ble skrevet i første utkastet. Grunnen var at det var store arbeidsmengder og et blankt ark som aktiverte begge medlemmene. Dette ledet til ekstremt effektivt arbeid. Ettersom prosjektet utviklet seg videre, så fant Erlend og Tord ut at finpussingsdelen og testing innebar mer sammensveiset koding, så ansvarsområdene slo seg mer sammen. Dette fungerte hovedsakelig bra, men det skjedde enkelt tilfeller med overlapp på grunn av dårlig kommunikasjon.
- **Stian** hadde hovedansvar for designet og byggingen av roboten. Hans fokus lå i å bygge en ressurssterk, minimalistisk og effektiv robot. I sin helhet ble roboten veldig godt bygget med en grundig struktur hvor hver eneste del av roboten var planlagt, og spilte en rolle i robotens funksjonalitet. Feiltakelsen her; var å ikke ha en fullstendig, tydelig visjon for hele teamet, fra første kloss var plassert. Dette ledet til noen små misforståelser mellom programmeringsteamet og Stian, slik at Erlend og Tord hadde noen misforståelser. Heldigvis ble dette tatt opp tidlig, og teamet opprettholdt en god arbeidsflyt.
- **Christian** Hadde hovedansvar for skriving, kvalitetssikring og delegering av rapportskriving. Fokuset hans lå i å sette opp, og skrive rapporten. Ettersom resten ble ferdige med sitt arbeid, deletere skriving av rapport til dem. Det som kunne vært gjort bedre her, var å skrive mer utdypende svar i rapporten fra start. Siden fokuset i teamet lå i konsis og effektivt arbeid, så ledet det til flere steder i teksten hvor videre skriving eller omskriving var nødvendig for å gi et utfyllende nok svar. Det hadde derfor spart en del tid og skrive fullstendig fra starten av. Tilfreds for dette har Christian delegert arbeidsmengden godt slik at rapporten har hatt en effektiv utvikling.

For å konkludere, så har ansvarsfordelingen fungert effektivt og bra, men med rom for forbedring. Kommunikasjonen mellom Erlend og Tord kunne vært bedre for å tydeliggjøre hvem som har ansvar for hva, slik at teamet kunne unngått de få tilfellene med kode overlapp. I tillegg kunne Stian vært bedre på å kommunisere visjonen til gruppen, og Christian kunne ha vært mer grundig i startprosessen i

rapportskrivingen. I alt, har teamet hatt tydelige ansvarsområder, hvor alle har forholdt seg til, og gjennomført sitt arbeid, men med muligheter for forbedring.

### **Bruk av samhandlingsverktøy.**

Teamet utnyttet hovedsakelig gitlab som samhandlingsverktøy. Gruppen oppdaget tidlig at versjonskontroll og samarbeid gjennom Git ville være en utfordring som måtte tas tak i, ettersom de fleste hadde begrenset erfaring med dette fra før. Det tok en stund før alle i gruppen ble komfortabel med de grunnleggende Git-operasjonene som «fetch», «pull», «add», «commit» og «push». Denne læringskurven antyder at en gjennomgang av Git-bruk sammen tidligere i prosjektet kunne vært hjelpsomt.

I tillegg, så teamet at for større prosjekter ville mer, og bedre organisert bruk av issue-board vært nyttig. Dette verktøyet kunne hjulpet med å lage en tydeligere struktur til prosjektet i sin helhet. Fordeling av oppgaver og arbeid, samt følge fremgangen til prosjektet hadde vært betydelig lettere med en kontinuerlig og god bruk av issue-boards. Teamet erkjenner å ikke utnytte det fulle potensialet til Git, noe som kunne gjort arbeidet mer strukturert og effektivt.

Hadde teamet anskaffet en solid forståelse for Git og implementert en mer systematisk og strukturert bruk av issue-board fra starten av, ville trolig ha gjort flere aspekt ved prosjektet med strømlinjet. Dette kunne ha spart tid og frustrasjon, spesielt i oppstartsfasen av oppgaven, der alt ofte er kaotisk og ustrukturert.

Denne erfaringen har belyst viktigheten av å finne og adressere potensielle utfordringer tidlig i prosessen, særlig når det gjelder samhandlingsverktøy. For fremtidige prosjekter vil teamet prioritere gjennomgang eller opplæring av sentrale prosjektverktøy fra begynnelsen, for å sikre en mer produktiv og effektiv arbeidsflyt for hele teamet.

### **Kommunikasjon**

Teamets sterkeste side har vært kommunikasjon, både mellommenneskelig og personlig, og profesjonelt og koordinerende. Det har vært effektiv og god bruk av teams for å dele egen situasjonsforståelse og for å stille spørsmål og få oppklaringer i forvirringsmomenter. I tillegg har alle på gruppen opptrådt høflig og imøtekommende med hverandre både i og utenfor timer. Denne gode kommunikasjonen har utvilsomt vært et nøkkelmoment for prosjektets suksess, og har også bidratt til trivsel blant medlemmene.

Selv om teamets sterkeste side har vært denne gode kommunikasjonen, så har teamet hatt enkelte steder med miskommunikasjon. Som sagt tidligere så hadde teamet enkelte tilfeller hvor programmeringsteamet

misforsto den fulle visjonen til utseende av roboten. Dette ledet til vanskeligheter for programmeringsteamet om sluttproduktet, som i sin helhet stagnerte arbeidsflyten deres.

Heldigvis ble dette problemet tatt opp relativt tidlig, og alle i teamet fikk en innføring i hvordan sluttproduktet burde se ut. Dette klarerte all problematikk om sluttproduktet for programmeringsteamet, og de kunne fortsette arbeidet sitt.

### Målbevissthet

Et viktig kriterium i alle teamarbeid er å jobbe mot et felles mål, og dette er definitivt noe som har kjennetegnet teamet. Alle ble tidlig enige om at det ikke er konkurranse som er drivende for teamet, og at det heller er et ønske om å lage et så godt produkt som mulig. Dette har vært en nøkkeltankegang for alle medlemmer gjennom hele prosessen, noe som har bidratt til et godt samarbeid og et helhetlig utformet produkt.

I tillegg har teamet vært flinke til å fastsette delmål på starten av hver eneste økt. Den interne gruppen har hatt morgenmøter, som kort tar opp hvordan hver del av teamet ligger an på sin arbeidsfront, samt hva neste steg er. Også har gruppen sett på hva neste mål for teamet som helhet er. Dette har sikret at teamet har jobbet på en strukturert måte, som har gjort hele teamet bevisst på hva som skal til for å nå delmålet sitt, samt hvor gruppen til slutt skal ende opp i sluttmålet.

### Planmessighet

I alle timer tok teamet en kjapp samling på starten for å etablere og fordele arbeidsoppgaver, men prioriteringen her ble ikke alltid fulgt da det var en tendens til å få tunnelsyn på sin tildelte/valgte oppgave. Her ser gruppen i ettertid at det kunne blitt fattet tiltak rundt delmål der fullføring av mål burde blitt sagt tidligere og heller flyttet fokus over på et annet område. Mangel på planmessighet har til syvende og sist ikke satt en stor stopper for arbeidet, men om det hadde vært belyst som et fokusområde i starten kunne det fortsatt ha bidratt til enda mer effektiv bruk av tid og ressurser.

### Tidsbruk

Innenfor tidsbruk har det vært generelt lite problemer da teamet har brukt timene effektivt og stort sett jobbet kontinuerlig med prosjekt. Det største problemet her har vært at det til tider har vært vanskelig å sysselsette alle gruppemedlemmene fordi det har vært for få ting å ta stilling til. Hadde dette vært identifisert tidligere kunne gruppen antagelig jobbet mer effektivt gjennom deler av prosjektet.

## Resultat

I bunn og grunn er teamet veldig fornøyd med resultatet. SpanX bærer preg av en helhetlig og god prosess enighet og godt samarbeid fra start til slutt. Det er selvfølgelig forbedringer som kunne bli gjort, men gitt forutsetningene og utgangspunktet teamet startet med, så er SpanX et vellykket produkt av et godt samarbeid. Skulle oppgaven blitt gjort på nytt ville vi ikke gjort endringer på sluttproduktet først, men heller raffinert prosessen. Her burde gitlab ha blitt utnyttet til en større grad, spesielt issue-boards for å få en mer strukturert og strømlinjet arbeidsprosess. Det skulle også ha blitt fokusert mer på å legge et sikkert fundament rundt områdene hvor misforståelser og feiltakelser oppsto, igjen for å sikre bedre flyt i arbeidsprosessen. En annen ting som trolig hadde gjort annerledes, hadde vært å gjennomgå alle deler av arbeidsprosessen mer utdypende for hele teamet. Dette slik at teamet kunne fungert mer modulært og effektiv om en på teamet skulle vært fraværende i løpet av arbeidsprosessen. Det hadde bygget en tryggere grunnstruktur og lagt opp en plan B om noen på teamet skulle vært vekke. Dette kunne gjort sluttresultatet enda mer raffinert med bakgrunn i at vi hadde hatt mer tid til produktutvikling og større kapasitet til samhandling og koordinering inn i resten av prosessen.

## Konklusjon og anbefalinger

Generelt sett er har prosjektet og arbeidet rundt SpanX vært veldig godt fremgang hos alle team-medlemmene, på grunn av god læring og deling av kunnskap som fokusområde. Teammedlemmene er alle fornøyde med produktet som er laget. Dette er fordi SpanX er resultatet av et grundig arbeid der alle har vært delaktige og har deltatt på plan- og beslutningsprosessen. Allerede fra brainstormingsfasen har gode diskusjoner fått frem alle sine meninger og synspunkt, noe som har bidratt til et bedre og mer omfattende læringsutbytte for alle teamets medlemmer.

Det har vært momenter hvor gruppen har sett forbedringspotensial, som beskrevet i diskusjonsdelen av rapporten. Deriblant så ble nytteverdien av Git forstått for sent, noe som ikke bare hadde kunnet øke produktiviteten i arbeidet, men som også hadde vært et område det kunne vært lurt å sette seg inn i ettersom det vil være relevant for flere fag senere og også nyttig i arbeidslivet. Rent samarbeidsmessig mener teamet at det er en kombinasjon av flaks og tilpasningsdyktighet som gjorde at arbeidet fungerte så bra som det gjorde, men noe læring er det jo selvfølgelig å hente der også.

## Bærekraft og samfunnspåvirkning

Prosjekt SpanX er designet med et klart mål: å møte samfunnsbehov, spesielt i krisesituasjoner som følge av naturkatastrofer. Denne innovative løsningen har potensiale til å gjøre en betydelig forskjell for mennesker og lokalsamfunn ved å tilby rask tilgang til mobilitet der det er mest nødvendig.

*Bidrag til FNs bærekraftsmål:*

- **Mål 9: Industri, innovasjon og infrastruktur**

SpanX representerer en nytenking innenfor infrastruktur. Med sin effektive evne til å plassere broer raskt i områder utsatt for flom, skred eller andre naturkatastrofer. Så bidrar prosjektet SpanX til å bygge solid infrastruktur. Dette er viktig for å fremme bærekraftig utvikling og innovasjon, og det støtter FNs bærekraftsmål; delmål 9.1 om pålitelig bærekraftig infrastruktur [1].

- **Mål 11: Bærekraftige byer og lokalsamfunn**

Når naturkatastrofer inntreffer, kan SpanX hjelpe med å gjenopprette mobiliteten og infrastrukturen raskt ved å sette opp midlertidige broer. Dette gjør byer og lokalsamfunn mer motstandsdyktige mot naturkatastrofer og andre kriser som påvirker infrastrukturen. I alt bidrar

SpanX til å redusere konsekvensene av katastrofer ved å sikre at samfunn kan opprettholde tilgang til nødvendige ressurser, som broer [1].

SpanX er utviklet med fokus på bærekraft fra bunnen av. Når det gjelder bærekraftig design, er SpanX bygget på en ressurseffektiv måte som utnytter et minimalistisk design. Selv om designet er minimalistisk, så fokuserer det på kvalitet og effektivitet. Hver eneste del av roboten er grundig gjennomtenkt slik at alle delene spiller en rolle, dette gjør roboten ekstremt ressurseffektiv, og understreker at fokuset er på bærekraft og kvalitet.

For å sikre bærekraftig implementering av brobyggerroboten SpanX, er det viktig å forstå konteksten og behovene i hvert bruksområde. Det er nyttig å lage strategier som tar hensyn til de unike utfordringene som kan oppstå, med tydelige planer for hva som skal gjøres, når det skal gjøres, og hvem som har ansvar for de ulike oppgavene. God ledelse som fremmer implementering er avgjørende – ledere som er løsningsorienterte, kunnskapsrike om systemet og utholdende gjennom prosessen er viktige for best mulig implementering av SpanX, spesielt i krisesituasjoner.

Videre er det essensielt å vurdere hvordan SpanX kan påvirke de samfunnene den skal hjelpe. Prosjektets evne til å tilby rask respons i krisesituasjoner kan redde liv og redusere økonomiske tap, men det bringer også spørsmål om rettferdighet i distribusjon av teknologien. Hvem skal få tilgang til SpanX? De som betaler mest, eller de som trenger det mest? Hvordan sikrer man at de mest utsatte samfunn prioriteres? Spesielt hvis de ikke har samme mengder summer som andre, som ofte er tilfellet i slike steder.

I tillegg må vurdering av etiske implikasjoner om datainnsamling og personvern også betraktes når SpanX er i felt. I krisesituasjoner er det kritisk med effektivitet og struktur, men til hvilken grad skal dette prioritertes overfor andres personvern? Det vil si at det er helt avgjørende at roboten ikke bare fokuserer på effektivitet, men også ansvarlighet overfor menneskene den påvirker.

Ved å innpasse bærekraftige designprinsipper samt gjennomtenkte implementeringsstrategier, har SpanX potensialet til å bidra til kraftigere og fleksible infrastrukturelle løsninger. SpanX systemet kan redusere økonomiske tap og sosiale utfordringer etter naturkatastrofer eller annen infrastrukturproblematikk. Ved å muliggjøre rask respons kan prosjektet:

- Redusere økonomiske tap og sosiale utfordringer etter naturkatastrofer
- Forbedre livskvaliteten i avsidesliggende samfunn ved å gi sikker tilgang til tjenester og ressurser
- Styrke lokalsamfunns evne til å håndtere fremtidige utfordringer ved å sikre at infrastruktur gjenopprettes raskt.

Prosjektet er en spennende tilnærming til bærekraftig infrastruktur gjennom bruk av en brobyggerrobot som adresserer viktige samfunns- og bærekrafts behov. Gjennom fokus på fleksibilitet, effektivitet og miljøvennlighet, bidrar SpanX til FNs bærekraftsmål, særlig innen infrastruktur og bærekraftige samfunn. Dette prosjektet har potensialet til å bidra med en substansiell positiv forskjell for samfunnet, gitt at de riktige etiske betraktingene er tatt, spesielt i krisetider som etter en naturkatastrofe.

## Bibliografi

- [1] FN, «FNs Bærekraftsmål,» 1 Februar 2024. [Internett]. Available: <https://fn.no/om-fn/fns-baerekraftsmaal>.
- [2] «LEGO MINDSTORM - EV3 Devices,» The LEGO Group, 2020. [Internett]. Available: <https://pybricks.com/ev3-micropython/ev3devices.html>. [Funnet 16 Oktober 2024].



## Vedlegg til prosjektrapporten

### Arbeidskontrakt (Vedlegg 1)

# Arbeidskontrakt for team 15

Medlemmer: Tord Johannessen Fosse, Christian Petter Remman, Erlend Sundsdal og Stian Closs Walmann

#### **Innledende tekst**

Denne arbeidskontrakten bygger på et sett med typiske mål, oppgavefordelinger, prosedyrer og retningslinjer for interaksjoner for studentarbeider. Arbeidskontrakten er utfylt med *egne* fortolkninger av hva man mener med disse og hvordan man skal oppnå dette.

Mål.....	1
Effektmål.....	1
Resultatmål .....	2
Roller og oppgavefordeling .....	2
Prosedyrer .....	3
Møteinnkalling.....	3
Interne møter .....	3
Eksterne møter.....	3
Varsling ved fravær eller andre hendelser .....	3
Dokumenthåndtering.....	3
Innleveringer av gruppearbeider .....	3
Interaksjon.....	4
Oppmøte og forberedelse.....	4
Tilstedeværelse og engasjement .....	4
Hvordan støtte hverandre.....	4
Uenighet, avtalebrudd .....	5

#### **Mål**

##### **Effektmål**

- Utvikle teamets kodeegenskaper
- Bli bedre på å samarbeide i team
- Kunne forholde seg til forskjellige samhandlingsverktøy
- Kjenne til og kunne benytte forskjellige metoder for møter og arbeidsrelatert kommunikasjon

- Utvikle enkeltindividet ved hjelp av teamets kunnskap og erfaring
- Benytte spesialistkompetanse der mulig og sørge for at kompetansen deles med hele teamet
- Vi skal praktisere godfotteorien. Dvs. vi skal gjøre de andre på teamet gode. Vi skal utnytte hverandres sterke sider og utvikle vårt team til å bli det perfekte individ
- Vi ønsker å utnytte gruppens ressurser på en best mulig og effektiv måte, slik at alle er i aktivitet til enhver tid og er kjent med den interne arbeidsfordelingen

### Resultatmål

- I løpet av prosjektets varighet skal hele teamet forstå koden bak roboten.
- Teamet skal møte opp til interne fredagsmøter, hvor alle skal si hva de har holdt på med og hva de skal gjøre ut dagen.
- I løpet av prosjektets varighet skal alle i teamet ha vært møtesjef, og ha full kontroll over all informasjon gitt ut i informasjonskanalene (teams, mail, osv..).
- I løpet av prosjektets varighet skal teamet kunne bruke og forstå alle samhandlingsverktøy (gitlab, teams, osv..) og forholde seg til arbeidet og til hverandre ved hjelp av disse verktøyene.
- Hvert teammedlem skal utvikle minst en ny ferdighet eller kompetanse innenfor hver av de forskjellige aspektene ved prosjektet (programmering, bygging, ledelse, osv..).
- I løpet av prosjektets varighet skal de som har spesialkompetanse fått bidra med dette til prosjektet, og at den anvendte kompetansen blir delt med resten av gruppa på en slik måte at personen kan gjenfortelle.

### Roller og oppgavefordeling

Vi har valgt en flat organisasjonsstruktur for vårt prosjektsamarbeid, noe som gir alle gruppemedlemmer like muligheter til å uttrykke seg og bli hørt. Før hver arbeidsøkt diskuterer vi oppgavene som må utføres og fordeler disse basert på teamets individuelle styrker. Alle oppgaver registreres og tildeles via GitLab's issueboard. Ved uenighet gjennomfører vi en avstemming i gruppen. Hvis vi fortsatt ikke klarer å bli enige, lar vi ChatGPT ta avgjørelsen for oss. Vi er også åpne for å revurdere vår organisasjonsstruktur og innføre en dedikert lederrolle dersom det viser seg å være mer hensiktsmessig for prosjektets fremgang.

Møtesjef og sekretær for eksterne møter planlegges intern ref. Eksterne møter --> Møteinkalling.

## Prosedyrer

### Møteinkalling

#### Interne møter

Møter skjer hver fredag morgen, med mindre annet er avtalt. Disse interne møtene er viktige for å dele oppdateringer, så det er essensielt at deltakerne kommer forberedt. Disse interne møtene kalles inn via Teams. Innkallinger sendes ut 2 dager i forveien, og det er viktig å bekrefte deltagelse raskt.

#### Eksterne møter

Eksterne møter, som involverer faglærer og studentassistent, innkalles via e-post til begge parter. Innkallingen skal sendes minst to dager før møtet. Alle eksterne møter er obligatoriske, men kan gjennomføres digitalt ettersom man har gitt beskjed til teamet, faglærer og studentassistent. Eksterne møter krever en møteleder som har ansvar for møteinkalling og en sekretær som har ansvar for møtereferat. Disse rollene rulleres internt i teamet og planlegges via Teams.

### Varsling ved fravær eller andre hendelser

Ved forventet forsinkelse eller fravær fra møter eller arbeid, skal teammedlemmet umiddelbart varsle via den dedikerte Teams-gruppen. Teamet vil deretter vurdere muligheten for digital deltagelse og avklare forventninger til den fraværende sitt bidrag og involvering. Dette sikrer effektiv kommunikasjon og minimal forstyrrelse av teamets arbeid.

### Dokumenthåndtering

GitLab brukes som vår plattform for dokumentlagring og versjonshåndtering, noe som sikrer at alle endringer spores og at teamet har tilgang til oppdaterte dokumenter. For effektiv samskriving registreres oppgaver på issueboards, som teammedlemmer må sjekke og oppdatere jevnlig. Dette forhindrer at oppgaver blir gjort flere ganger unødvendig og sikrer at alle er informert om pågående oppgaver.

### Innleveringer av gruppearbeider

Vi har fastsatt interne frister for innlevering av gruppearbeider som ligger tre dager før den offisielle fristen som gis av faglærer. Denne praksisen sikrer at vi har tilstrekkelig tid til å overholde de faktiske fristene og håndtere eventuelle uforutsette situasjoner.

## Interaksjon

### Oppmøte og forberedelse

Punktlig oppmøte er essensielt for effektivt gruppearbeid. Oppmøtetidspunkt følger generelt timeplanen, med mindre annet er avtalt. Alle forventes å møte forberedt med nødvendig utstyr og materialer. For eksempel skal møteleddere ha en klar agenda for møtet. Ved dagens start skal hvert gruppemedlem gi en kort statusoppdatering om:

- Hva de har jobbet med siden sist
- Planlagte oppgaver for dagen

Denne praksisen sikrer god kommunikasjon, fremmer ansvarliggjøring, og holder alle oppdatert på prosjektets fremgang. God forberedelse og punktlighet bidrar til produktive møter og effektivt gruppearbeid.

Sistemann som møter til møtet, tar ansvar for å starte de interne møtene.

### Tilstedeværelse og engasjement

Tilstedeværelse og engasjement er avgjørende for å nå våre mål. Dette innebærer oppmøte og aktiv deltagelse i møter og prosjekter. Vi forventer at alle bidrar med sin kompetanse, tar initiativ, og støtter hverandre. Under gruppearbeid skal alle være fokuserte på den aktuelle oppgaven og unngå distraksjoner eller sideaktiviteter. Gjennom denne dedikerte tilstedeværelsen og engasjementet skaper vi et produktivt arbeidsmiljø der alle trives og utvikler seg.

### Hvordan støtte hverandre

- **Åpen kommunikasjon** – Vi skal holde en åpen kommunikasjon med ærlig og konstruktiv tilbakemelding, slik at alle føler seg verdsatt. Vi holder tilbakemeldinger som tilbakemeldinger, og ikke mistolker dette som personlige angrep.
- **Samarbeid** – Vi skal støtte hverandre i oppgaver og prosjekter, dele kunnskap og gi hverandre rom for å stille spørsmål, hvor ingen spørsmål er dumme. Det skal skapes et miljø av respekt og lojalitet til hverandre slik at man kan stille hverandre spørsmål uten å føle seg dum.
- **Feiring av suksesser** – Vi vil anerkjenne og feire både teamets prestasjoner, uansett størrelse, for å styrke motivasjonen til hverandre.
- **Fleksibilitet** – Vi respekterer hverandres behov for balanse mellom arbeid og privatliv, og tilpasser arbeidsoppgaver der det er mulig. Dette betyr ikke at man kan være ”lat”, men at man føler en form for flyt og trygghet innad gruppen.

- **Sosiale aktiviteter** – Vi vil arrangere sosiale sammenkomster for å styrke relasjoner og bygge et godt felleskap.

### Uenighet, avtalebrudd

Uregelmessigheter skal diskuteres på teammøter, både med faglærer og internt. Avvik defineres primært som å ikke møte opp til gruppens bestemte tidspunkt uten forvarsel, grunn eller annen avtale med gruppen, eller å ikke være tilgjengelig via Teams eller lignende plattformer når fysisk oppmøte ikke er mulig. Ved første og andre avvik gis en muntlig advarsel fra teamet, og den berørte får mulighet til å forklare seg og foreslå en løsning. Ved tredje avvik sendes en skriftlig advarsel til både den berørte og faglærer, og saken diskuteres på neste teammøte. Ved fjerde avvik innkalles den berørte til en dialog med faglærer for å diskutere situasjonen og vurdere fremtidige konsekvenser, som kan inkludere utesettelse fra spesifikke prosjekter eller i ytterste konsekvens fra kurset. Alle avvik loggføres. Ved ekstraordinære omstendigheter kan faglærer vurdere mindre konsekvenser, og positiv forbedring over tid kan føre til at tidligere avvik "slettes" fra loggen. Forebyggende tiltak inkluderer jevnlig evaluering av teamdynamikk og individuell innsats.

Trondheim, 30. August. 2024

Tord Johannessen Fosse

Tord Fosse

Erlend Sundsal

Erlend Sundsal

Christian Petter Remman

Christian Petter Remman

Stian Closs Walmann

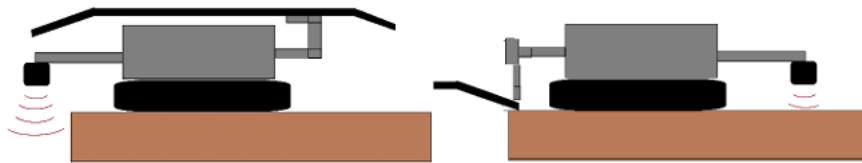
Stian Closs Walmann

## Prosjektplan (Vedlegg 2)

# Prosjektplan

## Problembeskrivelse

Vi har bestemt oss for å utvikle en robot som kan bygge midlertidige broer over elver og avrunner. Målet er å forbedre mobiliteten i områder uten infrastruktur, spesielt etter naturkatastrofer som jordskred eller flom, der tradisjonelle broer kan være ødelagt. Broleggeren skal være mobil, fleksibel og raskt kunne sette opp broer som tåler både mennesker og lettare kjøretøy. Ved å gjøre det enklere for hjelpefolk å nå frem til rammende områder, håper vi å bidra til en mer effektiv nødhjelp. Dette prosjektet gir oss en unik mulighet til å kombinere teknologi med praktisk problemløsning for å skape noe som virkelig kan gjøre en forskjell.



## Målbare og styrbare SMART mål

### Effektmål

- Demonstrasjon av potensialet for automatisert brobygging som bidrag til bærekraftig infrastruktur gjennom en fungerende prototype ved prosjektets slutt.
- Øke interessen for robotikk og programmering blant medstudenter gjennom prosjektets demonstrasjon og presentasjon.
- Utvikle gruppens ferdigheter innen prosjektledelse og teknisk implementering, målbart gjennom prosjektets gjennomføring og sluttrapport.
- Inspire til innovativ tenkning rundt automatisering i bygge bransjen, evaluert gjennom tilbakemeldinger fra medstudenter og faglærere etter presentasjonen.

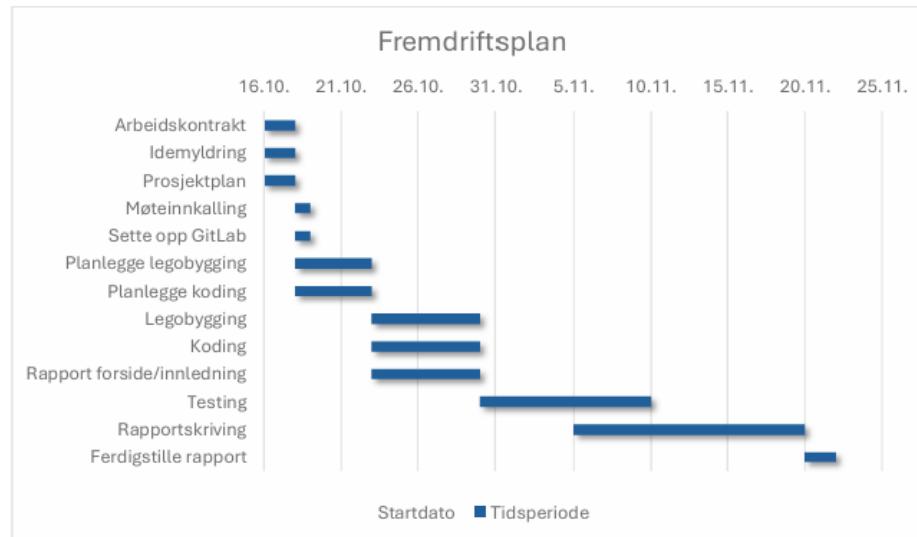
## Resultatmål

- Konstruere en funksjonell Lego Mindstorms-robot som kan bygge en enkel bro innen utgangen av uke 45.
- Utvikle et Java-program basert på objektorienterte prinsipper for styring av roboten, ferdig til testing innen slutten av uke 45.
- Utarbeide en omfattende prosjektrapport som dokumenterer utviklingsprosessen, ferdig innen slutten av uke 46.
- Presentere prosjektet og demonstrere roboten for medstudenter og faglærere i uke 47.
- Gjennomføre en testfase av roboten i uke 45, med dokumenterte resultater og forbedringer implementert i uke 46.
- Fullføre all koding og bygging innen utgangen av uke 45.
- Gjennomføre finpussing og ferdigstilling av prosjektet i løpet av uke 46.

## Prosessmål

- Gjennomføre daglige morgenmøter hver time for å evaluere fremdriften og tildele ansvar til hvert gruppemedlem for den aktuelle timen, samt ukentlige teammøter for overordnet prosjektstyring og planlegging.
- Sikre en jevn arbeidsfordeling mellom gruppemedlemmene gjennom hele prosjektperioden.
- Gjennomføre ukentlige kodegjennomganger for å forbedre kodekvalitet og kunnskapsdeling innad i gruppen.
- Utføre minst to testfaser av roboten med dokumenterte forbedringer mellom hver fase.

## Fremdriftsplan



## Risikovurdering

All risiko er temmelig neglisjerbar da det er snakk om plastikk på under 2 kilo som beveger seg ekstremt sakte. Likevel har vi identifisert noen punkter som kan være verdt å merke seg.

	Konsekvens			
	Sannsynlighet	Liten	Middels	Kritisk
Svært stor	■	■	■	■
Stor	■	■	■	■
Middels	■	■	■	■
Liten	■	■	■	■
Minimal	■	■	■	■

Faremoment	Konsekvens	Sannsynlighet	Risiko	Tiltak	Sannsynlighet	Restrisiko
Snublefare	Liten	Liten	1	Lyd og lys	Minimal	1
Batteribrann	Kritisk	Liten	2	Ikke lade uten oppsyn	Minimal	1
Feil bruk av bro	Middels	Liten	1	Merke broen	Minimal	1

## Kvalitetssikring

Under kodingen vil objektorientert kode brukt for å gjøre koden mer dynamisk, og lettere å jobbe sammen med uten at det ødelegger for resten av koden når endringer blir gjort.

I main branchen i gitlab vil det kun ligge kode som fungerer. Endringer og testing av kode som ikke nødvendigvis gjøres i andre branches og kan merges med main når det er sikkert at koden fungerer.

Vi skal ha tydelige rutiner for kvalitetssikring av koden:

- Når et medlem er ferdig å kode, skal et annet teammedlem kjapt se over koden, og kvalitetssikre.
- For hver «kodeøkt» skal branches lages for å sikre at «main» koden ikke overskrives med en feil, og for å ha god, og oversiktig struktur.
- Når merging av branches skjer, skal et annet teammedlem godkjenne og kvalitetssikre at koden er skrevet godt nok for å merges.
- Programmeringsansvaret i teamet, byttes mellom hver time for å sikre at koden får forskjellige innfallsvinkler, og ny tenking. Dette vil være svært hjelpende mot «tunnel vision» og gjøre koden mer dynamisk og fleksibel.
- På slutten av hver økt skal hele teamet gjennomgå koden og sikre at alt er slik det skal være

## Møteinkallinger og møtereferat (Vedlegg 3)

Møteinkalling 23.10.2024:

### Møteinkalling

Innkalling til møte 23.10 i Team 15

Trondheim, 18.10.2024

Møteinkallingen går til:

Christian Petter Remman, Stian Closs Walmann, Erlend Sundsdal, Tord Johannessen  
Fosse, Shiza Ahmad, Grethe Sandstrak

Tid og sted: Onsdag 18.10.2024 Klokka: 12:15-12:37(22min) Elektro-bygget, Møterom E304.

Møteleader blir Christian P. Remman  
Sekretær blir Erlend Sundsdal

#### Agenda

Saknr.	Saker	Tid	Beslutning	Ansvarlig person
01	Intro	1 min		Møteleader
02	Team Utvikling(Deltar alle? Teamutviklingsaktiviteter?)	4 min		Møteleader/Stian
03	Gjennomgang av prosjektplan	6 min		Møteleader/Tord
04	Status for prosjektet(valg av oppgave, fremdrift, aktivitetsdiagram, programmering	7 min		Møteleader
05	Evt.	3 min		Møteleader/Erlend
06	Avslutning	1 min		Møteleader

Ta forbindelse med Møteleader Christian P. Remman [chripre@stud.ntnu.no](mailto:chripre@stud.ntnu.no) om du ikke kan møte.

Velkommen.

Hilsen

Christian Remman

Møtereferat 23.10.2024

Saknr.	Saker	Tid	Beslutning	Ansvarlig person
01	Intro	1 min		Møteleder
02	Team Utvikling(Deltar alle? Teamutviklingsaktiviteter?)	4 min	Utviklet oss, god oppgavefordeling, alle er effektive. Stoler mer på hverandre, ikke mange aktiviteter utenom	Møteleder/Stian
03	Gjennomgang av prosjektplan	6 min	Er gode på:  Høre med andre. Friske/nye tanker Versjonskontroll. Branching. Bytting på oppgaver. Statussjekk.	Møteleder/Tord
04	Status for prosjektet(valg av oppgave, fremdrift, aktivitetsdiagram, programmering)	7 min	Kommet frem med programmeringen. Ferdig ish med bygging	Møteleder
05	Evt.	3 min	Hele prosjektet skal være i wikien, som dokumenter og slikt.  Møteinnkallinger kan legges inn som skjermbilder  Alt som har skjedd i prosjektet kan organiseres i fanner  ALT skal inn i wikien  Kan utflytte mer på tiltak under risikovurdering	Møteleder/Erlend
06	Avslutning	1 min		Møteleder

Se Prosjektplan vedlegg 2

Møteinkalling 13.11.2024

## Møteinkalling

Innkalling til møte 13.11 i Team 15

Trondheim, 08.11.2024

Møteinkallingen går til:

Christian Petter Remman, Stian Closs Walmann, Erlend Sundsdal, Shiza Ahmad,  
Grethe Sandstrak

Tid og sted: Onsdag 13.11.2024, kl: 12:15 i Gløshaugen Realfagsbygget Rom A4-112

Møteleder: Tord Fosse

Referent: Christian Remman

### Agenda

Saknr.	Saker	Tid	Ansvarlig person
01	Godkjenning av møteinkalling	1 min	Møteleder
02	Utført arbeid sist uke	4 min	Alle gruppemedlemmer
03	Kommende arbeid neste uke	4 min	Alle gruppemedlemmer
04	Finnes det en hindring for arbeid neste uke	4 min	Alle gruppemedlemmer
05	Klargjøringsrunde om standup møtet	4 min	Møteleder
06	Eventuelt	4 min	Møteleder
07	Avslutning	1 min	Møteleder

Meld fra til [tordjf@ntnu.no](mailto:tordjf@ntnu.no) dersom du ikke kan møte.

Mvh

Tord Fosse  
Møteleder

Møtereferat 13.11.2024

Møtereferat

Utført arbeid sist uke:

Tord – ~~kodeprosess~~ rapport

Stian – rapport

Erlend – finjustert programmet og begynt på rapport

Christian – rapport

Neste uke/fredag

Presentasjoner neste uke

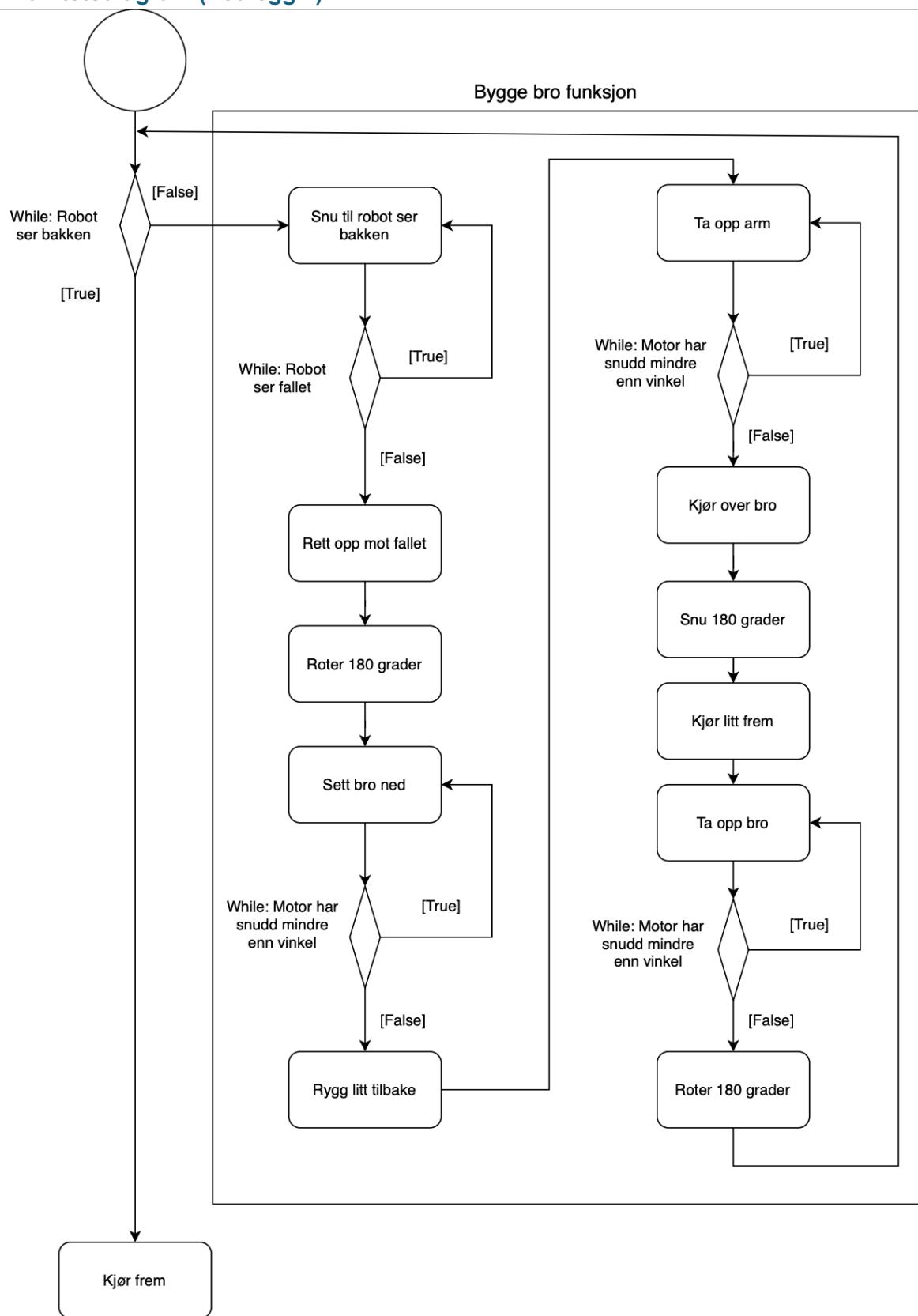
Tord – Presentasjon

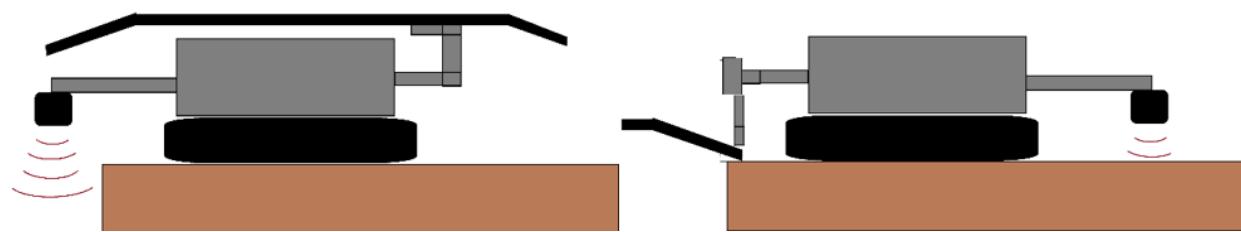
Erlend – Presentasjon

Stian – Presentasjon

Christian - Presentasjon

## Aktivitetsdiagram (Vedlegg 4)



**Skisse av utseendet til førsteutkast av roboten (Vedlegg 5)**

**Timeliste med statusrapport (Vedlegg 6)**

Person	Kategori	Aktivitet	Dato	Timer
Erlend	administrasjon	Skrive prosjektplan	16.10.2024	6
Erlend	programmering	opprette gitprosjekt	18.10.2024	4
Erlend	programmering	Broleggefunksjon	23.10.2024	6
Erlend	programmering	Broleggefunksjon/klasse	30.10.2024	6
Erlend	programmering	Optimering	01.11.2024	4
Erlend	dokumentasjon	skrive rapport	06.11.2024	4
Erlend	dokumentasjon	Skrive rapport	08.11.2024	4
Erlend	dokumentasjon	Skrive rapport	13.11.2024	6
Erlend	dokumentasjon	Skrive rapport/presentasjon	15.11.2024	4
Erlend	Dokumentasjon	Presentasjon, rapport finpuss	20.11.2024	6
				Totalt: 50
<hr/>				
<hr/>				
<hr/>				
Stian	Konstruksjon	Bygge hovedkonstruksjon	16.10.2024	8
Stian	Konstruksjon	Bygge hovedkonstruksjon	18.10.2024	4
Stian	Konstruksjon	Bygge kjørefunksjon	23.10.2024	6
Stian	Konstruksjon	Bygge Brofunksjon	25.10.2024	4
Stian	Konstruksjon	Bygge Brofunksjon	30.10.2024	6
Stian	Konstruksjon	Finpusse konstruksjon	01.11.2024	4
Stian	Dokumentasjon	Skrive rapport	06.11.2024	6
Stian	Dokumentasjon	Skrive rapport	08.11.2024	4
Stian	Dokumentasjon	Skrive rapport	13.11.2024	5
Stian	Annet	presentasjon og forberedelse	15.11.2024	4
Stian	Annet	Presentasjon, sortere lego	20.11.2024	4
				Totalt: 55
Tord	Administrasjon	Skrive prosjektplan	16.10.2024	6
Tord	Programmering	planlegging/ev3	18.10.2024	4
Tord	Programmering	Kjørefunksjon	23.10.2024	6
Tord	Programmering	Testing/Feilsøking	25.10.2024	4
Tord	Programmering	Testing/Feilsøking	30.10.2024	6
Tord	Programmering	kode ferdig hovedprogram	01.11.2024	4
Tord	Programmering	Feilsøking/finpusse	06.11.2024	6
Tord	Dokumentasjon	skrive kodeprosess	08.11.2024	4
Tord	Dokumentasjon	kodeprosess, om meg, retting	13.11.2024	6
Tord	Annet	presentasjon og forberedelse	15.11.2024	4
Tord	Annet	Presentasjon, rapport finpuss	20.11.2024	4
				Totalt: 54
Christian	Dokumentasjon	Skrive prosjektplan	16.10.2024	6

Christian	Dokumentasjon	Utarbeidelse av rapport	18.10.2024	4
Christian	Dokumentasjon	Utarbeidelse av rapport	23.10.2024	6
Christian	Dokumentasjon	Utarbeidelse av rapport	25.10.2024	4
Christian	Dokumentasjon	Utarbeidelse av rapport	30.10.2024	6
Christian	Dokumentasjon	Utarbeidelse av rapport	01.11.2024	4
Christian	Dokumentasjon	Utarbeidelse av rapport	06.11.2024	4
Christian	Dokumentasjon	Utarbeidelse av rapport	08.11.2024	4
Christian	Dokumentasjon	Utarbeidelse av rapport	13.11.2024	6
Christian	Annet	Presentasjonsforberedelse	15.11.2024	4
Christian	Annet	Presentasjon, rapport finpuss	20.11.2024	4
Totalt:				52

Uke 1

Timeliste	Stian	
Aktivitet	Kategori	Antall timer
Prototyping	Konstruksjon	8
Prototyping	Konstruksjon	4
#REF!		12.0

Timeliste	Christian	
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	6
Utarbeidelse av rapport	Dokumentasjon	4
#REF!		10.0

Timeliste	Erlend	
Aktivitet	Kategori	Antall timer
Administrasjon av eget arbeid	Skrive prosjektplan	6
Implementasjon - kildekode	opprette gitprosjekt	4
#REF!		10.0

Timeliste	Tord	
Aktivitet	Kategori	Antall timer
Administrasjon av eget arbeid	Skrive prosjektplan	6
Implementasjon - kildekode	planlegging/ev3	4
#REF!		10.0

Uke 2

Timeliste		Stian
Aktivitet	Kategori	Antall timer
Prototyping	Konstruksjon	6
Implementasjon - kildekode	Konstruksjon	4
#REF!		<b>10.0</b>

Timeliste		Christian
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	6
Utarbeidelse av prosjektrapport	Dokumentasjon	4
#REF!		<b>10.0</b>

Timeliste		Erlend
Aktivitet	Kategori	Antall timer
Implementasjon - kildekode	Programmering	6
#REF!		<b>6.0</b>

Timeliste		Tord
Aktivitet	Kategori	Antall timer
Prototyping	Programmering	6
Implementasjon - kildekode	Programmering	4
#REF!		<b>10.0</b>

## Uke 3

Timeliste		Stian
Aktivitet	Kategori	Antall timer
Implementasjon - kildekode	Konstruksjon	6
Feilretting av program	Konstruksjon	4
#REF!		<b>10.0</b>

Timeliste		Christian
Aktivitet	Kategori	Antall timer
Utarbeidelse av rapport	Dokumentasjon	6
Utarbeidelse av rapport	Dokumentasjon	4
#REF!		<b>10.0</b>

Timeliste		Erlend
Aktivitet	Kategori	Antall timer
Testing av egne program	Programmering	6
Feilretting av program	Programmering	4
#REF!		<b>10.0</b>

Timeliste		Tord
Aktivitet	Kategori	Antall timer
Feilretting av program	Programmering	6
Implementasjon - kildekode	Programmering	4
#REF!		<b>10.0</b>

Uke 4

Timeliste		Stian
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	6
Utarbeidelse av prosjektrapport	Dokumentasjon	4
#REF!		<b>10.0</b>

Timeliste		Christian
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	4.0
Utarbeidelse av prosjektrapport	Dokumentasjon	4.0
#REF!		<b>8.0</b>

Timeliste		Erlend
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	4.0
Utarbeidelse av prosjektrapport	Dokumentasjon	4.0
#REF!		<b>8.0</b>

Timeliste		Tord
Aktivitet	Kategori	Antall timer
Feilretting av program	Programmering	6.0
Utarbeidelse av prosjektrapport	Dokumentasjon	4.0
#REF!		<b>10.0</b>

Uke 5

Timeliste		Stian
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	5
Presentasjon med forberedelse	Dokumentasjon	4
#REF!		<b>9.0</b>

Timeliste		Christian
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	6.0
Presentasjon med forberedelse	Dokumentasjon	4.0
#REF!		<b>10.0</b>

Timeliste		Erlend
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	6.0
Presentasjon med forberedelse	Dokumentasjon	4.0
#REF!		<b>10.0</b>

Timeliste		Tord
Aktivitet	Kategori	Antall timer
Utarbeidelse av prosjektrapport	Dokumentasjon	6.0
Presentasjon med forberedelse	Dokumentasjon	4.0
#REF!		<b>10.0</b>

Uke 6

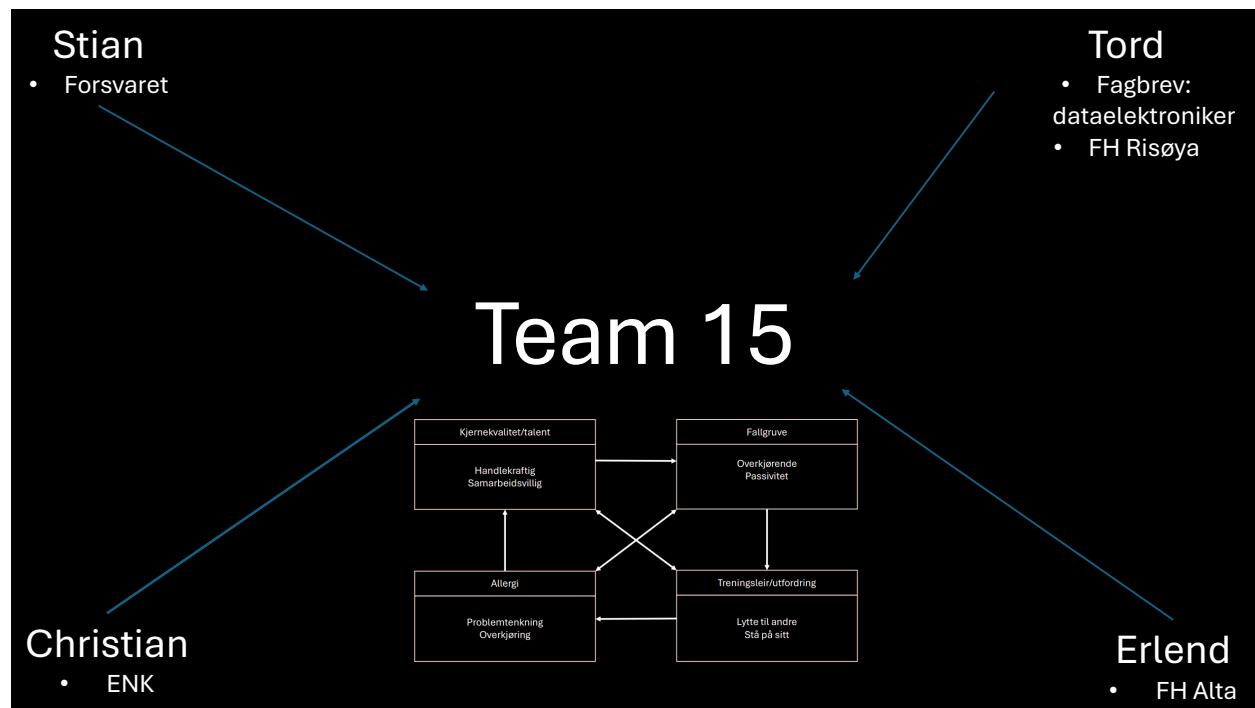
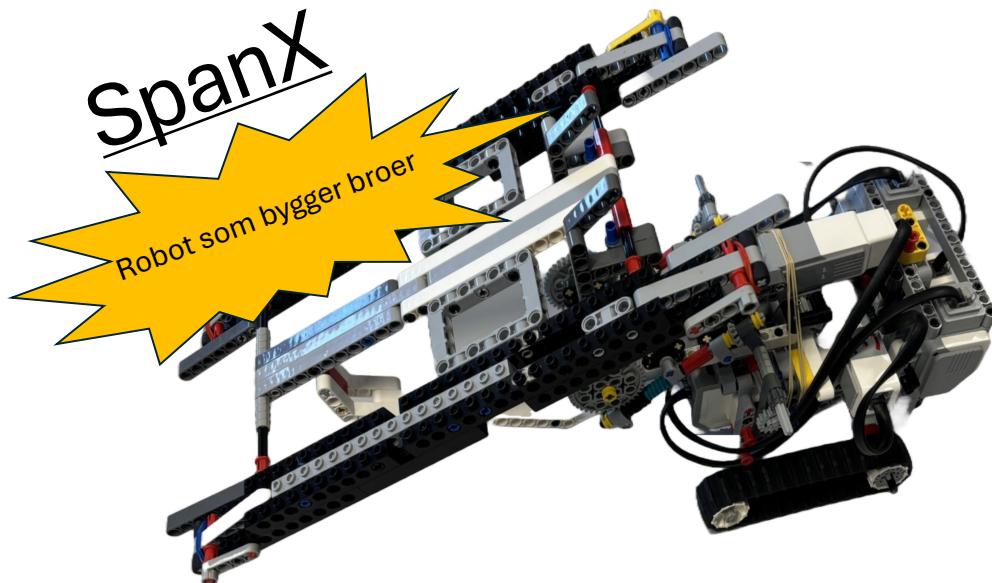
Timeliste		Stian
Aktivitet	Kategori	Antall timer
Presentasjon med forberedelse	Dokumentasjon	4
#REF!		4.0

Timeliste		Christian
Aktivitet	Kategori	Antall timer
Presentasjon med forberedelse	Dokumentasjon	4
#REF!		4.0

Timeliste		Erlend
Aktivitet	Kategori	Antall timer
Presentasjon med forberedelse	Dokumentasjon	4
#REF!		4.0

Timeliste		Tord
Aktivitet	Kategori	Antall timer
Presentasjon med forberedelse	Dokumentasjon	4
#REF!		4.0

## Presentasjon av prosjektet (Vedlegg 7)





## Prosess

Legobygging – Stian

Rapport – Christian

Koding – Erlend og Tord

Kort briefing ved start

# Utfordringer

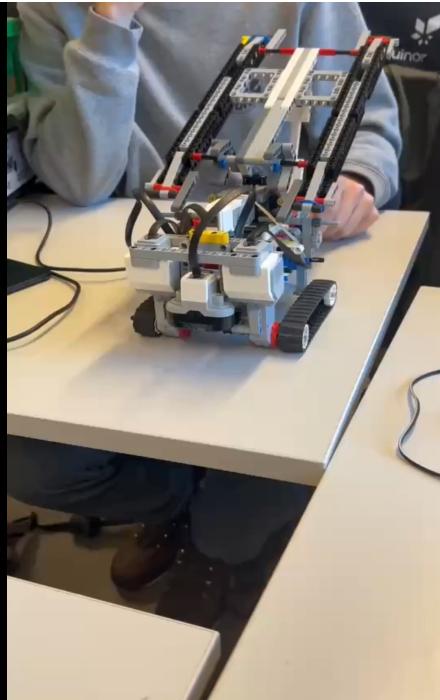
Git

Sensorer

Koding

Rapport

Demo



# Spørsmål?

Team 15

## Refleksjonsnotat (Vedlegg 8)

---

### Refleksjonsnotat

Prosessen i team 15 har jevnt over vært bra. Vi har hatt en flat struktur i teamet som har gjort at alle har følt på ansvar i forhold til produktet som leveres.

Arbeidsfordelingen har falt ganske naturlig på plass ettersom alle hadde felter de følte seg mer komfortable med.

I starten var vi bekymret for om det kom til å bli kollisjon mellom de forskjellige personlighetene i teamet, men ettersom vi belyste denne muligheten tidlig så følte det til at dette ikke ble et problem.

Teamet hadde en flat struktur der alle hadde like mye ansvar. Vi delte arbeidsoppgavene slik:

- Stian Walmann – Ansvar for legobyggingen og design av robot  
Bidratt med erfaring innenfor legobygging. Lagde et avansert kransystem for broleggeren. Store deler av rapporten er også skrevet av Stian.
- Erlend Sundsdal – Ansvar for koding av brobyggerklasse  
Bidratt med å opprette gitprosjektet i GitLab. Lagde kode for kranfunkasjonene til roboten. Feilsøkt og testet kode med bruk av roboten. Har også bidratt til en mer utfyllende rapport. Skrevet på forord og ordnet med vedlegg.
- Christian Remman – Ansvar for prosjektrapporten  
Bidratt med god erfaring innenfor rapportskriving. Skrevet store deler av rapporten. Skrevet: Sammendrag, abstract, introduksjon, problembeskrivelse, mye av resultater og diskusjon, og «bærekraft og samfunnspåvirkning».
- Tord Fosse – Ansvar for koding av driveklasse og mainklasse  
Bidratt med koding av driveklasse og mainklasse for roboten. Sammenslått og dokumentert koden, samt feilsøkt og testet kode. Har også skrevet om kodeprosessen i rapporten.

Vi tror teamprosessen har vært så bra som den har vært fordi alle i teamet har vært imøtekommende og tilpasningsdyktige.

I tillegg har det vært god og profesjonell kommunikasjon innad som har ført til at alle har kunne gjøre seg hørt og forstått. Dette har minsket faren for misforståelser og miskommunikasjon som har bidratt til både effektivitet og trivsel.

I tillegg har arbeidsfordeling og ledelse vært delt mellom alle medlemmene likt, noe som har gjort at alle føler seg inkludert i styring- og beslutningsprosessen. Dette har ført til at alle har eierskap og ~~og~~ ansvar på lik linje og ingen har sluntret unna.

Denne oppgaven har vært veldig givende og lærerik. Det største læringsmomentet har vært å jobbe fra start til slutt i et team, og ende med et reelt produkt. Denne prosessen har vært svært lærerik fordi det er en helt reel tilnærming til teamarbeid. Vi har sammen måtte løse problemer og uforutsigbarheter sammen, noe som har gjort at vi har knytt godt bånd og blitt venner. Som helhet har teamet lært (faglig perspektiv):

- Samhandling med nye mennesker
- Hvordan jobbe i team
- Hvordan håndtere ulikheter og uenigheter
- Hvordan skape resultater fra en ide til et produkt
- Arbeidsdelegering
- Jobbe uavhengig av hverandre, men mot samme mål

Fra et personlig perspektiv har vi lært:

- Knytte bånd med helt tilfeldig nye mennesker
- Respektere og lytte til andre sine meninger og ideer
- Være behjelpeig når andre sliter med arbeidsoppgavene sine
- Planlegge og prioritere oppgaver
- Objektorientert koding i ~~python~~

Dette har vært et vellykket prosjekt hvor alle i teamet har funnet ut og lært både nytt om seg selv, samt mye faglig – spesifikt hvordan samhandle i et team.

Dersom vi kommer i en tilsvarende situasjon senere vil vi gjøre mye av det somme som vi allerede har gjort i dette prosjektet. Arbeidsmetoden og arbeidsfordelingen har fungert godt. Versjons-kontrollert kode, og arbeid i objekt-orientert kode gjør det effektivt å jobbe i team med samme oppgave. Tydelig kommunikasjon fra start med standupmøter og statusrapporter underveis gjør prosessen oversiktlig da alle medlemmene jobber mot samme mål. Dette vil vi også fortsette med.

Kort sagt har vi tatt mye lærdom av det å jobbe i team og hvordan man håndterer ulikheter og delegering av oppgaver.

Kommunikasjonen har vært avgjørende for teamet vårt. Det har blitt tydelig kommunisert hva vi kan, hvilke egenskaper vi har og hvordan de har bidratt inn i teamet.

For vår del har det fungert bra med en flat struktur uten en klar lederrolle, men det kan være det har vært et resultat av flaks mer enn ferdigheter. Det skal uansett sies at teamets tilpasningsdyktighet har spilt en stor rolle i å få dette til å fungere.

Nytten av objektorientert programering og versjons kontrollert kode er svært effektivt i teamarbeid.