

Helina Kitsing (Tartu Ülikool), 2011



Euroopa Liit
Euroopa Sotsiaalfond



Eesti tuleviku heaks

E-kursuse
„Robotika projekt“
materjalid

Aine maht 9 EAP

Helina Kitsing (Tartu Ülikool), 2011

Kursuse tutvustus

Kursuse eesmärk on omandada praktilisi oskusi elektroonikas, mehaanikas, mehatronika projekti juhtimises, disainis ning meeskonnatöös. Kursus toimub intensiivse meeskonnatöö vormis, kus piiratud aja jooksul tuleb koostada robotikaalane projekt ning see ellu viia.

Õppeaine läbinud üliõpilane:

- oskab eesmärgistada robotika või muu riistvaraarenduse projekti, määratleda selle oodatavat tulemust ning koostada ülevaatlikku projektiplaani ja eelarvet
- oskab jagada projekti alamtöödeks ja määrata nende vahelised sõltuvused ning teostatavuse järjestuse, hinnata tegevuste mahtusid ning tulemustele vastavust
- oskab dokumenteerida projekti tegevusi ning suuliselt ja kirjalikult edasi anda erialast infot ja põhjendada oma seisukohti projekti osapooltele
- suudab osaleda meeskonnatöös nii tavaliselt kui ka projektijuhina ning on võimeline tagama tööde tähtaegse valmimise
- tajub probleeme, mis on seotud projektidega, millel on mitu täideviivat osapoolt, ning oskab vajadusel tekkinud probleeme lahendada
- oskab vastavalt plaanile projekti ellu viia ning kohandada tegevusi vastavalt projekti nõuete muutumisele

Kursuse tegevuskava

- Nädal 1: Meeskondade moodustamine ja meeskonna analüüs
- Nädalad 2, 7: Projekti ettevalmistavad tegevused (projekti eesmärgistamine, nõuete kogumine, projekti planeerimine - suuremate vaheetappide määratlemine)
- Nädal 3, 13: Projekti abistavad tegevused (suhtlemine kliendiga, sponsorluse otsimine)
- Nädal 4: Meeskonnatöö korraldamine (rollide ja vastutuse jagamine, töökorraldus ja abivahendid, meeskonnatöö reeglid)
- Nädal 5 - 6: Projekti lahenduse täpsustamine (lahenduse valik, riskide määratlemine ja riskide maandamine, eelarve koostamine)
- Nädal 8 - 10: Mehaanika, elektroonika planeerimine, disainimine, realiseerimine. Tarkvara planeerimine, disainimine, programmeerimine.
- Nädal 11 - 12: Mehaanika, elektroonika ja tarkvara testimine ning parandamine.
- Nädal 14: Töö tulemuste analüüs
- Nädal 15 - 16: Töö aruandlus ja esitlus

Iseseisev töö: Põhiosa ainekst moodustab iseseisev töö materjalidega ja ülesannete lahendamine. Igaks nädalaks on antud ülesanded, mis tuleb ära lahendada ja esitada läbi Moodle'i. Ülesannete lahendamiseks on igal nädalal abiks materjalid. Ülesannete esitamine tähtjast hiljem on lubatud, kuid maksimaalne hinne selliste esituste eest on 50% võrra väiksem.

Hindamisele pääsemise tingimus: Kõik ülesanded peavad olema esitatud ja saanud positiivse hinde (arvestatud või A, B, C, D, E).

Hinde kujunemine: Aine lõpphinne kujuneb hindeliste ülesannete hinnete (A, B, C, D, E) keskmisena.

E-õppe tegevused: Kursuse materjal on jaotatud 16 õppenädala peale ja see on mõeldud suuremas osas iseseisvaks õppimiseks. Teoreetilised materjalid sisaldavad kogu infot, mis on kursusel nõutud. Ülesanne lahendused tuleb esitada läbi Moodle'i.

Lisainformatsioon: alvo.aabloo@ut.ee

Selle kursuse kodulehe loomist on toetanud Euroopa Liit.

Õpiplaan

Nädal 1

Teema: Sel nädalal tuleb tutvuda aine korraldusega, moodustatakse meeskond projekti elluviimiseks ja analüüsitakse meeskonda - selgitatakse välja meeskonna oskused, tugevused ja nõrkused projekti elluviimise seisukohalt. Lisaks tutvutakse projekti ülesande püstitusega.

Oodatavad tulemused:

1. Moodustatud on meeskond (lahendatud ülesanne - *meeskonna wiki*)
2. Meeskonnal on selge aine korraldus (loetud materjal - *aine tutvustus*)
3. Meeskonnal on selge projekti ülesande püstitus (loetud materjal - *projekti kirjeldus*)
4. Meeskonna analüüs (loetud materjal - *meeskonna analüüs*, lahendatud ülesanne - *meeskonna analüüs*)

Materjalid:

- Aine tutvustus (PDF dokument)
- Projekti kirjeldus (PDF dokument)
- Meeskonna analüüs (PDF dokument)

Ülesanded:

- Ülesanne – meeskonna wiki
- Ülesanne – meeskonna analüüs

Nädal 2

Teema: Vaatame lähemalt, mida kujutab endast üks projekt, milline on projekti elutsükel ja kuidas see projekt jaguneb etappideks. Vaatame, millised on põhilised projektide ebaõnnestumise põhjused ja leiame võimalusi nende vältimiseks.

Oodatav tulemus:

1. Meeskonnal on selge, mis on projekt, projekti elutsükel ja etapid. Meeskond teab põhilisi probleemide ebaõnnestumise põhjuseid ja oskab neid vältida (loetud materjal - *projekt*)
2. Meeskond oskab kirjeldada oma projekti elutsükli, oskab jagada seda etappideks (lahendatud ülesanne - *projekti etapid*)
3. Meeskond oskab tuua välja probleemid, mille tõttu projekt võib ebaõnnestuda ning teab, kuidas neid vältida (lahendatud ülesanne - *projekti õnnestumine*)

Materjalid:

- Projekt (PDF dokument)

Ülesanded:

- Ülesanne – projekti etapid
- Ülesanne – projekti õnnestumine

Nädal 3

Teema: Selgitame, millised on põhilised probleemikohad klientidega suhtlemisel ja kuidas neid vältida. Vaatame, mis on projekti nõuded, milleks neid vaja on ja kuidas nõudeid kirja panna.

Oodatav tulemus:

- Meeskond teab, kuidas suhelda klientidega (loetud materjal - *kliendisuhtlus*)
- Meeskond oskab rahulikult ja mõistlikult lahendada probleemseid olukordi klientidega suhtlemisel (lahendatud ülesanded - *kliendiga suhtlemine (olukord 1)*, *kliendiga suhtlemine (olukord 2)*, *kliendiga suhtlemine (olukord 3)*)
- Meeskond teab, mis on projekti nõuded (loetud materjal - *nõuded*)
- Meeskond oskab nõudeid kirja panna (lahendatud ülesanne - *nõuded*)

Materjalid:

- Nõuded (PDF dokument)
- Kliendisuhtlus (PDF dokument)

Ülesanded:

- Ülesanne - nõuded
- Ülesanne - kliendiga suhtlemine (olukord 1)
- Ülesanne - kliendiga suhtlemine (olukord 2)
- Ülesanne - kliendiga suhtlemine (olukord 3)

Nädal 4

Teema: Vaatame, milliseid vahendeid kasutada meeskonnatöö korraldamiseks. Paneme paika meeskonnatöö reeglid ja jagame rollid ning vastutusvaldkonnad meeskonnas.

Oodatav tulemus:

1. Meeskond teab, kuidas korraldada meeskonnatööd (loetud materjal - *meeskonnatöö korraldamine*)
2. Meeskond oskab korraldada oma tööd (lahendatud ülesanne - *meeskonnatöö korraldamine*)
3. Meeskond teab, kuidas kasutada tasklisti (loetud materjal - *tasklist*)
4. Meeskond oskab jagada ülesandeid, kasutades tasklisti (lahendatud ülesanne - *tasklisti täitmine*)

Materjalid:

- Meeskonnatöö korraldamine (PDF dokument)
- Tasklist (PDF dokument)

Ülesanded:

- Ülesanne - meeskonnatöö korraldamine
- Ülesanne - tasklisti täitmine

Nädal 5

Teema: Tegeleme projekti lahenduse kavandamisega - teeme kavandi roboti lahenduse kohta, arvestades olemasolevate ressurssidega.

Oodatav tulemus:

- Meeskond teab, kuidas lahendust kavandada (loetud materjal - *lahenduse kavandamine*)
- Meeskond oskab kavandada lahendust oma projekti jaoks (lahendatud ülesanded - *lahenduse üldkirjeldus, mehaanika lahenduse kirjeldus, elektroonika lahenduse kirjeldus, tarkvara lahenduse kirjeldus*)

Materjalid:

- Projekti lahenduse kavand (PDF dokument)

Ülesanded

- Ülesanne - lahenduse üldkirjeldus
- Ülesanne - mehaanika lahenduse kirjeldus
- Ülesanne - elektroonika lahenduse kirjeldus
- Ülesanne - tarkvara lahenduse kirjeldus

Nädal 6

Teema: Analüüsime olemasolevat lahendust, leiame suuremad lahenduse riskid ja otsime võimalusi nende riskide maandamiseks. Täpsustame valitud lahendust ja koostame projekti eelarve.

Oodatav tulemus:

1. Meeskond teab, kuidas analüüsida oma projekti (loetud materjal - *projekti lahenduse analüüs*)
2. Meeskond oskab analüüsida oma projekti, leiab üles projektiga seotud riskid ja oskab neid maandada (lahendatud ülesanne - *lahenduse analüüsimine*)
3. Meeskond teab, kuidas koostada oma projekti kohta eelarvet (loetud materjal - *eelarve koostamine*)
4. Meeskond oskab oma projekti jaoks koostada eelarvet (lahendatud ülesanne - *eelarve koostamine*)

Materjalid:

- Projekti lahenduse analüüs (PDF dokument)
- Eelarve koostamine (PDF dokument)

Ülesanded:

- Ülesanne - lahenduse analüüsimine
- Ülesanne - eelarve

Nädal 7

Teema: Vaatame projekti tööde planeerimist, vaheetappide määratlemist ja eesmärkide seadmist. Koostame ülevaatliku projektiplaani.

Oodatav tulemus:

1. Meeskond teab, kuidas koostada ülevaatlikku projektiplaani (loetud materjal - *ülevaatlik projektiplaani*)
2. Meeskond oskab koostada ülevaatlikku projektiplaani (lahendatud ülesanne - *ülevaatlik projektiplaani*)

Materjalid:

- Ülevaatlik projektiplaani (PDF dokument)

Ülesanded:

- Ülesanne – ülevaatlik projektiplaani

Nädal 8

Teema: Elektroonika, mehaanika ja tarkvara valmistamine. Meeskond alustab tööd vastavalt oma projektiplaanile.

Oodatav tulemus:

1. Meeskond on täitnud ära eesmärgid esimeseks etapiks vastavalt oma projektiplaanile.
2. Meeskond oskab disainida elektroonikat (lahendatud ära ülesanded elektroonika disainimise kohta)
3. Meeskond teab, kuidas elektroonikaskeeme korralikult vormistada (loetud materjal - *elektroonikaskeemide joonistamine*)

Materjalid:

- Elektroonikaskeemide joonistamine (PDF dokument)

Ülesanded:

- Ülesanne - elektroonikaskeemide disainimine (AVR32, draiverite plaat)
- Ülesanne - elektroonikaskeemide disainimine (mootorite plaat)
- Ülesanne - elektroonikaskeemide disainimine (Coil-gun)

Nädal 9

Teema: Elektroonika, mehaanika ja tarkvara valmistamine. Meeskond jätkab tööd vastavalt oma projektiplaanile.

Oodatav tulemus:

1. Meeskond on täitnud ära eesmärgid teiseks etapiks vastavalt oma projektiplaanile.
2. Meeskond teab, kuidas joota elektroonikat (vaadatud videoloengud jootmise kohta)
3. Meeskond oskab joota elektroonikat (joodetud kokku oma projekti elektroonika)
4. Meeskond oskab programmeerida oma roboti riistvara (lahendatud ülesanded AVR-i kohta)

Materjalid:

- Jootmine 1 (FLV videoaken)
- Jootmine 2 (FLV videoaken)
- Jootmine 3 (FLV videoaken)

Ülesanded:

- Ülesanne – AVR-iga tutvumine
- Ülesanne – AVR.1
- Ülesanne – AVR.2
- Ülesanne – AVR32
- Ülesanne - jootmine
- Ülesanne - suhtlusprotokoll

Nädal 10

Teema: Elektroonika, mehaanika ja tarkvara valmistamine. Meeskond jätkab tööd vastavalt oma projektiplaanile.

Oodatav tulemus:

1. Meeskond on täitnud ära eesmärgid kolmandaks etapiks vastavalt oma projektiplaanile.
2. Meeskond teab, kuidas söövitada trükkplaate (vaadatud videoloengud söövitamise kohta)
3. Meeskond oskab koostada oma roboti mehaanikat (koostatud on roboti mehaanika).
4. Meeskond oskab juhtida elektroonikat (lahendatud ära ülesanded elektroonika kohta).

Materjalid:

- Mehaanika mudeli koostamine (fail)

- Söövitamine 1 (FLV videoaken)
- Söövitamine 2 (FLV videoaken)

Ülesanded:

- Ülesanne – Roboti mehhaanika mudel
- Ülesanne – mootori juhtimine
- Ülesanne – MLX90316SPI kasutamine
- Ülesanne - brushlessid

Nädal 11

Teema: elektroonika, mehaanika, tarkvara testimine ja parandamine.

Oodatav tulemus:

1. Meeskond teab, kuidas debugida oma roboti elektroonikat ja kasutab neid teadmisi oma roboti elektroonika debugimisel ja parandamisel.

Materjalid:

- Elektroonika debugimine 1 (FLV videoaken)
- Elektroonika debugimine 2 (FLV videoaken)
- Elektroonika debugimine 3 (FLV videoaken)

Ülesanded:

- Ülesanne – elektroonika debugimine

Nädal 12

Teema: elektroonika, mehaanika, tarkvara testimine ja parandamine

Oodatav tulemus:

1. Meeskond testib oma robotit, parandab vead ja paneb roboti tööle (lahendatud ülesanded roboti sõitmise kohta)

Ülesanded:

- Ülesanne – lihtne sõitmine
- Ülesanne – mööda väljaku servasid
- Ülesanne – lõpmatuse sõitmine
- Ülesanne – pallini sõitmine

Nädal 13

Teema: Tegeleme teemaga - kuidas leida sponsorit ja hankida sponsorlust.

Oodatav tulemus:

1. Meeskond teab, kuidas otsida sponsorit ja kavandada sponsorile esitlust (loetud matejal - *sponsorlus*)
2. Meeskond oskab leida oma sponsorit (lahendatud ülesanne - *sponsori hankimine*)

Materjalid:

- Sponsorlus (PDF dokument)

Ülesanded:

- Ülesanne – sponsori hankimine

Nädal 14

Teema: Käesoleva nädala teemaks on tulemuste analüüsimine. See on oluline etapp iga projekti juures - tulemuste analüüs annab meile eelised ja teadmised järgmiste projektide planeerimisel ja ellu viimisel.

Oodatav tulemus:

1. Meeskond teab, kuidas tulemusi analüüsida (loetud materjal - *tulemuste analüüsimine*)
2. Meeskond oskab analüüsida oma projekti tulemusi (lahendatud ülesanne - *tulemuste analüüs*)

Materjalid:

- Tulemuste analüüs (PDF dokument)

Ülesanded:

- Ülesanne – tulemuste analüüs

Nädal 15

Teema: Projektide dokumenteerimine, aluseks võtame ühe näidisaruande.

Oodatav tulemus:

1. Meeskond kogub kokku oma projekti käigus tehtud tööd ja vormistab need korrektse dokumendina (lahendatud ülesanne - *aruanne*)

Materjalid:

- Aruande näidis (PDF dokument)

Ülesanded:

- Ülesanne - aruanne

Nädal 16

Teema: Projektide esitlemine.

Oodatav tulemus:

1. Meeskond oskab esitleda oma projekti tööd (lahendatud ülesanne - *esitus*)

Ülesanded:

- Ülesanne - esitus

Kursuse Robotika projekt materjalide loetelu

Videoloengud

- Elektroonikaskeemide jootmine 1. osa, Jaas Ježov
- Trükkplaatide söövitamine 1.osa, Joel Kuusk
- Trükkplaatide söövitamine 2.osa, Joel Kuusk
- Elektroonikaplaatide debugimine 1.osa, Mihkel Pajusalu
- Elektroonikaplaatide debugimine 2.osa, Mihkel Pajusalu
- Elektroonikaplaatide debugimine 3.osa, Mihkel Pajusalu

Tekstmaterjalid (PDF dokumendid)

- Aine tutvustus
- Projekti kirjeldus
- Meeskonna analüüs
- Projekt
- Nõuded
- Kliendisuhetus
- Meeskonnatöö korraldamine
- Tasklist
- Projekti lahenduse kavand
- Projekti lahenduse analüüs
- Eelarve koostamine
- Ülevaatlik projektiplaan
- Elektroonikaskeemide joonistamine
- Sponsorlus
- Tulemuste analüüs

Ülesanded

- Meeskonna wiki
- Meeskonna analüüs
- Projekti etapid
- Projekti õnnestumine
- Nõuded
- Kliendiga suhtlemine (olukord 1)
- Kliendiga suhtlemine (olukord 2)
- Kliendiga suhtlemine (olukord 3)
- Meeskonnatöö korraldamine
- Tasklisti täitmine
- Lahenduse üldkirjeldus
- Mehaanika lahenduse kirjeldus
- Elektroonika lahenduse kirjeldus
- Tarkvara lahenduse kirjeldus

- Lahenduse analüüsimine
- Eelarve
- Ülevaatlik projektiplaan
- Elektroonikaskeemide disainimine (AVR32, draiverite plaat)
- Elektroonikaskeemide disainimine (mootorite plaat)
- Elektroonikaskeemide disainimine (Coil-gun)
- AVR-iga tutvumine
- AVR.1
- AVR.2
- AVR32
- Jootmine
- Suhtlusprotokoll
- Roboti mehhaanika mudel
- Mootori juhtimine
- MLX90316SPI kasutamine
- Brushlessid
- Elektroonika debugimine
- Lihtne sõitmine
- Mõõda väljaku servasid
- Lõpmatuse sõitmine
- Pallini sõitmine
- Sponsori hankimine
- Tulemuste analüüs
- Aruanne
- Esitlus

Kursuse LOFY.03.076 – Robotika projekt kirjeldus

Kursuse eesmärk on omandada praktilisi oskusi elektroonikas, mehaanikas, mehhatroonika projekti juhtimises, disainis ning meeskonnatöös. Kursus toimub intensiivse meeskonnatöö vormis, kus piiratud aja jooksul tuleb koostada robotikaalane projekt ning see ellu viia.

Õppeaine läbinud üliõpilane:

- oskab eesmärgistada robotika või muu riistvaraarenduse projekti, määratleda selle oodatavat tulemust ning koostada ülevaatlikku projektiplaani ja eelarvet.
- oskab jagada projekti alamtöödeks ja määrata nendevahelised sõltuvused ning teostatavuse järjestuse, hinnata tegevuste mahtusid ning tulemustele vastavust.
- oskab dokumenteerida projekti tegevusi ning suuliselt ja kirjalikult edasi anda erialast infot ja põhjendada oma seisukohti projekti osapooltele.
- suudab osaleda meeskonnatöös nii tavaliikme kui ka projektijuhina ning on võimeline tagama tööde tähtaegse valmimise.
- tajub probleeme, mis on seotud projektidega, millel on mitu täideviivat osapoolt, ning oskab vajadusel tekkinud probleeme lahendada.
- oskab vastavalt plaanile projekti ellu viia ning kohandada tegevusi vastavalt projekti nõuete muutumisele.

Kursuse tegevuskava:

- Nädal 1: Meeskondade moodustamine ja meeskonna analüüs
- Nädalad 2, 7: Projekti ettevalmistavad tegevused (projekti eesmärgistamine, nõuete kogumine, projekti planeerimine - suuremate vaheetappide määratlemine)
- Nädal 3, 13: Projekti abistavad tegevused (suhtlemine kliendiga, sponsorluse otsimine)
- Nädal 4: Meeskonnatöö korraldamine (rollide ja vastutuse jagamine, töökorraldus ja abivahendid, meeskonnatöö reeglid)
- Nädal 5 - 6: Projekti lahenduse täpsustamine (lahenduse valik, riskide määratlemine ja riskide maandamine, eelarve koostamine)
- Nädal 8 - 10: Mehaanika, elektroonika planeerimine, disainimine, realiseerimine. Tarkvara planeerimine, disainimine, programmeerimine.
- Nädal 11 - 12: Mehaanika, elektroonika ja tarkvara testimine ning parandamine.
- Nädal 14: Töö tulemuste analüüs
- Nädal 15 - 16: Töö aruandlus ja esitlus

Eeldusained

Kursusel osalemiseks on rangelt soovituslikud järgmised eeldusained:

- MTAT.03.100 Programmeerimine
- LOFY.03.009 Digitaalne loogika
- LOFY.03.014 Mikroprotsessorid
- LOFY.03.033 Robotika praktikum
- LOFY.03.016 Riistvaraprojekt
- MTAT.03.174 Projektijuhtimine
- MTAT.03.260 Pilditöötluse ja kujundituvastuse alused

Töökorraldus

- Töö toimub 4-6 liikmelistes meeskondades
- Igal nädalal toimub 1 seminar, mille jooksul meeskonnad annavad ülevaate oma töö seisust. Ülejäänud kursus on veebipõhine – igal nädalal tuleb meeskondadel tutvuda õppematerjalidega ja lahendada selleks nädalaks antud ülesanded keskkonnas Moodle. Täpsed juhised on kirjas Moodle-s.
- Igal meeskonnal on kohustuslik hoida oma projekti kohta käivat infot Moodle'is meeskonna wikis. Küsimuste esitamiseks ja aruteludeks on olemas üldine foorum, mida näevad kõik meeskonnad ning meeskonnasiseste teemade arutamiseks meeskonna privaatne foorum.

Hindamine

Hindamisele pääsemise tingimus: Kõik ülesanded peavad olema esitatud ja saanud positiivse hinde (arvestatud või A, B, C, D, E).

Hinde kujunemine: Aine lõpphinne kujuneb ülesannete hinnete keskmisena. Lõpphinne määratakse järgmisel: vähem kui 50% annab F, 50% - 59.9% annab E, 60%-69.9% annab D, 70%-79.9% annab C, 80 - 89.9% annab B ja alates 90% annab A.

Projekti kirjeldus

Ehitada jalgpallirobotite paar, mis suudab võistelda robotjalgpallis järgmiste mängureeglite alusel:

Mängitakse ühe oranži golfipalliga, mis on mängu alguses paigutatud väljaku keskele. Robotid stardivad oma väljaku poole nurkadest. Värava kaitsealas tohib korraga olla maksimaalselt üks kaitsev robot. Mängu jooksul on ühes võistkonnas olevatel robotitel lubatud omavahel suhelda.

Mäng koosneb kahest voorust, ühe voo ru kestvus on 5 minutit. Voorude vahel toimub poolte vahetus. Iga väravasse sisenenud pall annab ühe punkti. Kui pall siseneb väravasse, siis kohtuniku märguande peale (kohtunik vilistab) mäng ja mänguaeg seisatakse. Pall asetatakse uuesti väljaku keskele ja robotid oma väljaku nurkadesse stardipositsioonidele. Kohtuniku märguande peale stardivad robotid uuesti väljaku nurkadest ning mäng ja mänguaeg jätkub.

Mängu jooksul ei tohi robotid kahjustada üksteist ega võistlusväljakut.

Väljak, millel robotid võistlevad, vastab Robotex 2011 võistlusväljaku tingimustele (<http://www.robotex.ee/node/1384#Võistlusväljak>) ja robotid peavad vastama Robotex 2011 võistlusel robotitele esitatavatele nõuetele ([http://www.robotex.ee/node/1384#Robotile esitatavad nõuded](http://www.robotex.ee/node/1384#Robotile%20esitatavad%20nõuded)).

Meeskonna analüüs

Iga projekti tulemus sõltub väga palju seda ellu viivast meeskonnast. Sellepärast on tähtis analüüsida meeskonda – teha selgeks meeskonna oskused ja võimalused projekti seisukohalt lähtudes.

Meeskonnaliikmete oskused

Meeskonna analüüsi esimene etapp on meeskonnaliikmete oskuste välja selgitamine. Iga meeskonnaliikme kohta tuleks pingereas välja tuua oskused, mis on projekti seisukohalt kasulikud ja vajalikud. Pingereas võiks olla vähemalt 5 oskust, alustades nendest, milles meeskonnaliige on kõige pädevam, tuleb olla konkreetne.

Näiteks kui meeskonnaliige Juku oskab hästi SMD elektroonikakomponente plaadile joota, siis tulebki kirjutada „oskab joota SMD komponente“, mitte jääda üldiseks ja öelda, et Juku „oskab elektroonikat“. Detailselt välja toodud oskused aitavad hiljem meeskonnal jagada rolle ja vastutusvaldkondi ning valida paremini projekti jaoks sobivat lahendust.

Liiga üldiselt kirja pandud oskused võivad hiljem tekitada probleeme. Kui Juku oskuste pingereas on esikohal „oskab elektroonikat“, siis võib meeskond hiljem hätta sattuda, kui selgub, et Juku ei ole suuteline täitma talle antud elektroonika valdkonna ülesandeid (näiteks „elektroonikaplaadi disainimine“).

Meeskonna tugevused ja nõrkused

Kui meeskonnaliikmete oskused on välja selgitatud, tuleb edasi vaadata meeskonna üldpilti – leida meeskonna tugevused ja nõrkused. Tulemuse võiks ülevaate saamiseks vormistada tabelina. See tabel on kasulik hiljem projekti lahenduse valikul.

Näiteks kui meeskonnas nõrkuseks on elektroonikaplaatide disainimise oskus, siis tuleks robotiprojekti lahenduses kasutada võimalusel elektroonika valmislahendusi (näiteks Arduino vms). Nii tugevuste kui nõrkuste osas võiks kirja panna ligikaudu 10 tugevust ja 10 nõrkust.

Mõistlik on need tugevused ja nõrkused grupeerida valdkondade kaupa. Robotika projekti seisukohalt võiksid kindlasti olla esindatud järgmised valdkonnad: elektroonika, mehaanika, riistvara programmeerimine, tarkvara programmeerimine.

Näide meeskonna analüüsi tabelist:

Valdkond	Meeskonna tugevused	Meeskonna nõrkused
Elektroonika	<ul style="list-style-type: none">• Tugevus 1 (oskus joota SMD elektroonikakomponente)• Tugevus 2• ...	<ul style="list-style-type: none">• Nõrkus 1 (puudub oskus disainida elektroonikaplaate)• Nõrkus 2• ...
Mehaanika	<ul style="list-style-type: none">• ...	<ul style="list-style-type: none">•

Meeskonna analüüs

Võttes aluseks meeskonna tugevuste ja nõrkuste tabeli, tuleb läbi viia meeskonna analüüs. Hea olukord kompetentsi mõttes on siis, kui meeskonnas on igas vajalikus valdkonnas kirjas piisavalt tugevusi ja vähe nõrkusi ning esindatud on tugevused kõigis valdkondades. Probleemid tekivad kui:

- Meeskonnas on palju nõrkusi, kuid vähe tugevusi – võimaluse korral leida meeskonda juurde liige, kes lisab meeskonnale tugevusi vajalikes valdkondades.
- Meeskonnas on palju tugevusi, kuid kõik jäävad ühte valdkonda (näiteks elektroonika) – võimalusel vahetada mõni meeskonnaliige välja teistsuguste oskustega (mehaanika, programmeerimine) inimese vastu. Kui see pole võimalik, siis tuleb jagada meeskonnaliikmete vahel ära valdkonnad, milles on oskused/tugevused puudu, et iga meeskonnaliige saaks vastavas valdkonnas õppida ja oma oskusi projekti jaoks kasvatada.

Projekt

Käesolevas materjalis tegeleme järgmiste küsimustega:

1. Mis on projekt?
2. Milline on projekti elutsükkel?
3. Kuidas jagada projekti etappideks
4. Millised on põhilised projektide ebaõnnestumise põhjused?

Mis on projekt?

Vabas vormis definitsioon võiks kõlada nii: *Projekt on ettevõtmine, kus grupp inimesi püüab olemasolevate ressursidega ja piiratud ajaga saavutada ühist eesmärki.*

Antud aine raames:

- Grupp inimesi – 4-6-liikmeline meeskond
- Ühine eesmärk – ehitada robotite paar, mis vastab etteantud tingimuste ja suudab koos edukalt mängida 2 versus 2 robotjalgpalli.
- Olemasolevad ressursid – meeskonna teadmised ja oskused, laboris kasutatavad töövahendid, meeskonna eelarve
- Piiratud aeg – kursuse kestvus. Projekt peab olema valmis kursuse lõpuks.

Et täpsemalt aru saada, mida selle kursuse raames tehakse ja mida oodatakse, vaatame kõigepealt lähemalt ühe projekti elutsükli. Et see kõik ei jääks väga teoreetiliseks, vaatame näitena kõrvale üht reaalset projekti.

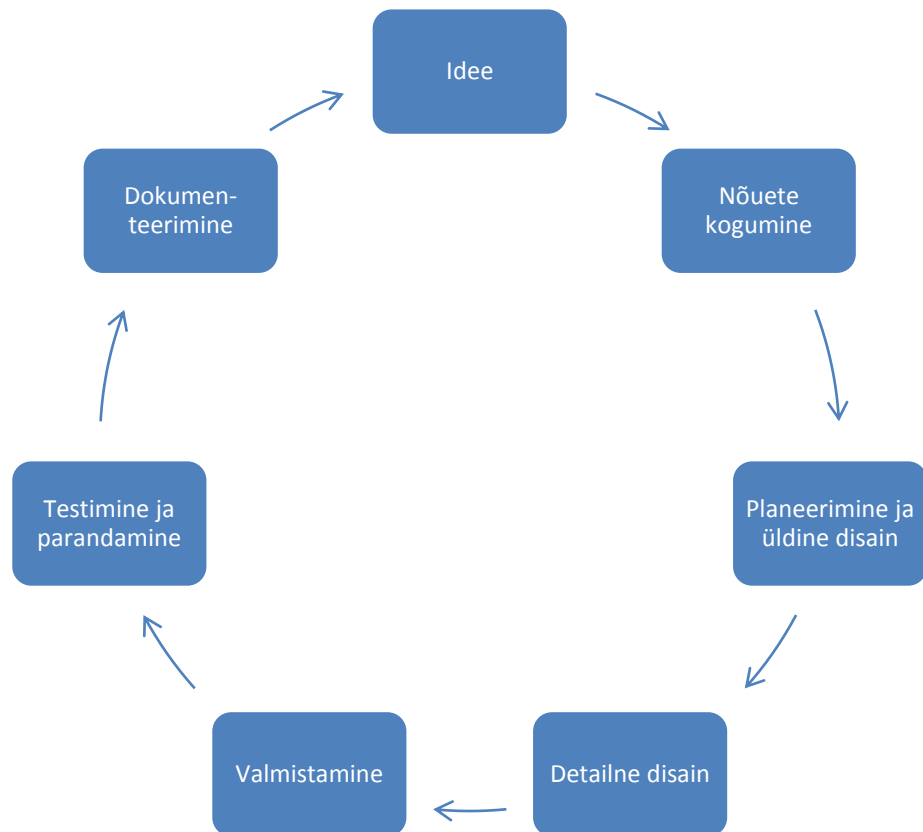
Projekti elutsükkel

Projekti „elu“ võime jagada erinevateks faasideks:

- **Idee faas.** Kõikide projektide alguseks on kellegi hea idee teha midagi ägedat, mis oleks mingil viisil kasulik – säästaks aega, raha, teeks kellegi rikkaks – või oleks lihtsalt äge. Aga idee üksi ei ole veel projekt. Paljud inimesed arvavad ekslikult, et kui on idee, siis see idee ongi see suur varandus ja sellest tuleb kümne küünega kinni hoida. Tegelikult aga – niikaua kui see idee jääbki ainult kellegi pähe ja realiseerimata, ei ole sellel praktiliselt muud väärtust kui olemas olemine.
- **Nõuete kogumine.** Idee faasile järgneb nõuete kogumise faas. Selle käigus tehakse selgeks, millised on täpsed vajadused ja tingimused ehitatavale süsteemile.
- **Planeerimine ja üldine disain.** Kui nõuded on kogutud ja projektile esitatavad tingimused täpselt paika pandud, alustatakse töö planeerimist. Planeerimise faasis tegeletakse põhiliselt järgmiste küsimustega:
 - Kuidas võiks nõuetele vastavat süsteemi ehitada? Kas üldse on võimalik niisugust ideed realiseerida või tuleb teha nõuetes järeleandmisi?
 - Mida ja kui palju on vaja, et projekt ellu viia? Siin vaadatakse nii materiaalseid ja rahalisi vahendeid kui ka inimressursse.
 - Kui kaua läheb aega, et projekt ellu viia? Kui palju aega on võimalik kulutada?
 - Millised on projekti juures riskid ja kuidas neid maandada?

- Jne
- **Detailne disain.** Kui üldine plaan ja süsteemi kontseptsioon on paigas, tuleb hakata realselt tööd tegema – disainima süsteemi mehaanikat, elektroonikat ja programme.
- **Valmistamine.** Kui disain on paigas, hakatakse süsteemi ehitama.
- **Testimine ja parandamine.** Valmis süsteemi testitakse ja parandatakse/täiendatakse nii, et see vastaks kõigile kokkulepitud nõuetele ja oleks kasutuskõlblik.
- **Dokumenteerimine.** Korraliku projekti juurde kuulub dokumentatsioon. Kirja tuleb panna kõik vajalik projektiga seoses. Olenevalt projektist hõlmab dokumentatsioon erinevaid dokumente. Vahel on piisavaks dokumentatsiooniks ainult süsteemi üldkirjeldus, mehaanika joonised, elektroonika skeemid ja programmi dokumentatsioon. Samas kui projektiks on midagi, mida kavatakse müüa, siis kuuluvad projekti juurde ka näiteks seadme kasutus- ja hooldusjuhendid jms.

Diagrammil võib projekti elutsüklit kujutada nii:



Näide

Vaatame näitena sellist väikest projekti: <http://hackaday.com/2011/10/16/we-want-this-led-ball/>

Antud projekti puhul võiksid faasid jaguneda järgmiselt:

- **Ideefaas:** Tuleb idee – teeks ägedaid elektroonilisi jõulukaunistusi
- **Nõuete kogumine:** Oletame, et teeme projekti ainult endale kasutamiseks, siis on ka nõuetelega lihtsam – tahame, et need kaunistused näeksid välja nagu jõulukuuse ehtimiseks kasutatavad kuulid ja tekitaksid huvitavaid valgusefekte – sarnaselt kuuskede ehtimiseks kasutatavate elektrikuunaldega.

- Planeerimine ja üldine disain: Mõtleme välja, et võiks elektroonikaplaadid teha kaarja kujuga, need omavahel ühendada kuuliga sarnaseks moodustiseks, sisse teha oma toiteallika ja LED- id kinnitada plaatide servadele. Siin faasis uuritakse ka võimalusi valmistamise jaoks – kust saaks disainitud elektroonikaplaate tellida, et ei peaks kõiki söövutama ja plaadid näeksid ilusad välja. Projekti valmistamisest piisab ainult ühe inimese (idee autori) teadmistest/oskustest ja tema enda rahalistest vahenditest. Töö toimub „viitsimise“ ajagraafiku alusel – siis kui idee autoril on aega ja tahtmist projektiga tegeleda ning projektil ei ole kindlat valmimise tähtaega (tüüpiline hobiprojektide puhul, aga mitte kommertsprojektide puhul).
- Detailne disain: Siin faasis toimub elektroonikaskeemide disainimine. Disainimise käigus valmistatakse ka mõni testskeem ja kontrollitakse, et kõik töötaks nii nagu peab. Samamoodi pannakse paika ka programmi arhitektuur.
- Valmistamine: Kui disain on paigas ja töötab, siis valmistamise faasis tellitakse vajalik kogus plaate ja pannakse kuulid kokku. Kirjutatakse programm valmis valgusefektide tekitamiseks.
- Testimine ja parandamine: Kuul on koos, programm on peal, siis pannakse kuul tööle ja kontrollitakse, et kõik töötab nii nagu vaja. Kui tulemusega ollakse rahul, loetakse test õnnestunuks.
- Dokumenteerimise faas: Kuna projekt on lihtne ja tehtud enda tarbeks, siis on dokumentatsioonis kirjas lihtsalt hädavajalik selleks, et teist korda samasugust kuuli teha oskaks. Laiemale maailmale tutvustamise jaoks on tehtud lehekülg internetti.

Üldiselt praktikas need faasid ajalises järjestuses kattuvad – testimine toimub paralleelselt valmistamise faasiga ja dokumenteerimine võib toimuda juba projekti alguses.

Projekti jagamine etappideks

Antud aine raames tegeleme projekti elutsükli faasides järgmiste teemadega:

1. Idee faas - Ülesande püstituse selgitamine, eesmärkide paika panek.
1. Nõuete kogumise faas – nõuete kogumine ja dokumenteerimine.
2. Planeerimine ja üldine disain - nõuetele tuginedes lahenduse kavandi koostamine. Lahenduse kavandis vaadatakse eraldi täpsustavalt mehaanikat, elektroonikat, programmi. Vaatame kavandatava lahenduse eelarvet. Projektiplaani koostamine lähtudes lahenduse kavandist, ressurssidest ja piirangutest (aeg), oodatavate tulemuste määramine. Meeskonnatöö planeerimine, korraldamine, järelvalve.
3. Detailne disain – lahenduse analüüs tuginedes nõuetele, olemasolevatele ressurssidele (meeskonna võimalused, rahalised jm vahendid) ja piirangutele (põhiliselt ajalised piirid), lahenduse korrigeerimine ja täpsustamine. Mehaanika ja elektroonika disainimine, programmi arhitektuuri kavandamine.
4. Valmistamise faas – projekti ellu viimine mehaanika, elektroonika ja tarkvara realiseerimine.
5. Testimine ja parandamine. Valminud lahenduse testimine ja parandamine. Tulemuste analüüsimine – vaatame, kuidas projekt valmis, lähtuvalt eesmärgist, nõuetest, plaanist ning oodatavatest tulemustest.
6. Tulemuste dokumenteerimine, esitlemine

Kui vaadata projekti elutsükli faase, siis on näha, et eduka elluviimiseks on vaja rohkem kui ainult teadmisi programmeerimisest või oskust elektroonikaskeeme disainida ja joota. Kui see ära unustada, siis paratamatult kipuvad projektid ebaõnnestuma.

Projektide ebaõnnestumise põhjused

Robotika projektide puhul kogemus näitab, et väga sageli projektid ei saa valmis õigeaks ajaks, lähevad maksma kordades rohkem kui rahalisi vahendeid olemas on või ei tööta nii nagu vaja. Mõned põhjused, miks nii juhtub:

1. **Puudulik või olematu planeerimine** – täpselt ei kavandata ette tehtavaid töid, jäetakse vahele planeerimise faas mõttega: „ma umbes tean, kuidas teha.. noh umbes kuu ajaga võiks valmis saada küll, ega seal midagi keerulist ei ole“. Tegelikult aga kulub lahenduse realiseerimiseks tunduvalt rohkem aega, sest kõiki töid ei osata ette näha või ei peeta piisavalt oluliseks, et nende tegemiseks aega arvestada. Näiteks väidetakse: „juhtmete lõikamine ja puhastamine võtab ainult pool tundi, see pole ju midagi.“ Järgmiseks väidetakse: „ega see paari juhtme plaadi külge jootmine ei võta ka rohkem kui 15 minutit“ jne. Kui need pisikesed tööd kokku võtta, kulub nende peale juba mõni päev. Ja kui selliseid projekti arvestamata tööpäevi koguneb liiga palju, siis pole võimalik jõudagi rumalast peast kokku lepitud lühikese ajaga valmis.
2. **Võimete ja oskuste ülehindamine** – arvatakse, et asjad käivad kõik lihtsalt, ilma probleemideta ja hinnatakse oma tööde tegemiseks kuluvat aega tunduvalt lühemaks, kui see tegelikult on. Näiteks mõeldakse: „see programm on ju ainult mõni rida, see ei võta rohkem kui paar õhtut.“ Tegelikult aga kulub programmi kirjutamise peale 2 päeva, programmist vigade otsimise ja parandamise peale veel 5 päeva. Ja kokku läheb aega tunduvalt rohkem, kui alguses arvatud.
3. **Töögraafiku puudumine** – sageli robotika projektide korral on kuulda väiteid: „elektroonika pole veel valmis, programmeerijad ei saa midagi teha“ või „elektroonikat ei saa teha, sest mehaanika pole paigas“. Tegelikult saab üldiselt igas projektis teha elektroonikat, mehaanikat ja programmeerimist paralleelselt. Väide: „ei saa teha, sest midagi muud on veel tegemata“ on viide halvale tööde planeerimisele ja kui järele mõelda, siis tuleb üldiselt alati välja, et kõike saab teha paralleelselt.
4. **Otsustamisega viivitamine** – projektis peab olema kindel piir, millal peavad tähtsad otsused olema tehtud (näiteks mis mootoreid kasutatakse või kui suuri akusid või millist kaamerat). Ei maksa nende otsustega viivitada. Kui projektis tuleb juba paar korda ette olukord, kus mingid tööd seisavad selle taga, et robotil kasutatavad komponendid on valimata, siis tuleks tegeleda nende valimisega ja ära otsustamisega, mitte jääda ootama, et „küll me seda tööd teeme hiljem, siis kui mootorid on valitud.“
5. **Liiga keeruka lahenduse valimine** – projekti puhul on oluline see, et tulemus oleks töökindel ja rahuldaks ette antud nõudeid, isegi siis, kui see tähendab igavama ja standardlahenduse valmistamist. Uusi asju on huvitav teha, aga kui selle hinnaks on projekti ebaõnnestumine, siis tuleb valida lihtsam ja ebahuvitavam kuid vähemalt valmis saav ja töötav lahendus. See on eeldus, et ka kunagi hiljem on võimalik teha teistsuguseid projekte. Kui ühe projektiga ebaõnnestutakse, siis uute projektidega teie jutule enam ei tulda.
6. **Plaanidest mitte kinni pidamine** – kui lahendus valitakse välja, lepitakse kokku ja alustatakse tööd, siis tuleb plaanidest kinni pidada. On rumalus (kahjuks liiga tihti esinev rumalus), et töö käigus avastatakse midagi uut ja huvitavat ja otsustatakse see uus ja huvitav kohe kasutusse

võtta. Üldjuhul lisab see projekti keerukust ja võib tekitada hiljem probleeme, mida algselt ei osata ette näha ja mille tõttu projekti pärast õigeaegselt ei lõpetata.

7. **Meeskonnatöö puudumine** – meeskond alustab tööd, lepatakse kokku, et iga liige teeb ära oma osa ja küll natuke enne tähtaja saabumist asjad kokku pannakse. Tegelikult probleemid just siis algavadki, kui on vaja erinevad süsteemi osad koos tööle panna. Probleemide lahendamine aga ei lähe kiiresti ja projekt ei saa õigeaegselt valmis.
8. **Alustamisega venitamine** – liiga tihti tuleb ette, et kui projekti lõpptähtaeg on kaugel, siis mõeldakse: „ah, küll ma homme alustan, täna on muudki teha. Nagunii on tähtaeg alles kaugel.“ Elu näitab, et tähtaeg jõuab kätte alati ootamatult. Sellepärast tuleb töödega alustada kohe.
9. **Ajutiste lahenduste kasutamine** – kuna ei taheta kohe korralikke lahendusi teha ja soovitakse mingit tulemust näha, siis tihti juhtub, et robotile kruvitakse rattad külge vildakalt ja elektroonika jäetakse isoleerimata vms. Ja pärast esialgse hooletuse tõttu põleb robotil maha pool elektroonikast või robot lihtsalt ei tee seda, mida programmi järgi peaks tegema. Siis juba parandatakse programmi, kuigi õige oleks lihtsalt rattad korralikult ja ilma loksumiseta roboti külge kinnitada. Tuleks vältida ajutisi ja lohakaid lahendusi – teha kohe korralikult. See võib alguses võtta pool tundi kauem, aga säästab hiljem mõned päevad lohakusest tekkinud probleemide lahendamisest.

Käesoleva aine raames üritame edukalt oma projektid lõpuni viia ja tüüpvigu vältida.

Projekti nõuded

Käesolev materjal üritab anda vastused järgmistele küsimustele:

1. Mis on nõuded?
2. Milleks nõudeid vaja on?
3. Kuidas nõudeid kirja panna?

Mis on nõuded?

Lühidalt - nõuded on tingimused, millele projekt peab vastama. Vaatame olukorda, kus meie juurde tuleb esindaja firmast X ja ütleb: „Meil on suured kontoriruumid ja öösel täiesti tühjad. Me mõtlesime, et võiks panna sinna robottolmuimeja, mis öösel kontorit koristab, nii et hommikul oleksid kõik ruumid puhtad.“

Siit saame oma projekti - ehitada robottolmuimeja. Aga lihtsalt ühest lausest projekti alustamiseks ei piisa ja küsime lisainfot. Kliendilt saame vastuseks: „Me soovime, et robot suudaks sõita iseseisvalt ruumides, ilma, et oleks juhtmega seinas. Robot võiks ka aru saada, kuidas see ruumis liiguks, et see esemete otsa ei takerduks. Töötada võiks näiteks öösel kui kedagi kontoris ei ole - võib rahulikult müristada. Robot peaks andma märku sellest, kui tolmukott täis on, et seda vahetada jne.“

Need kliendi soovid on nõuete alus ja nõuded on nö. „töödeldud variant“ kliendi soovidest. Tundub lihtne? Vaatame teemat natuke lähemalt ja selgub, et kõik päris nii lihtne ei olegi.

Milleks nõuded vaja on?

Üsna tihti algavad projektid nii, et kliendil on olemas visioon sellest, mida ta soovib, aga see visioon on üsna üldine – tahaks robotit, mis koristaks öösel kontorit. Kui seda visiooni mitte täpsustada ja mitte nõudeid kirja panna, alustada projekti tegevustega ja hiljem klient näeb pooltöötavat või töötavat robotit, siis enamasti:

1. klient pettub, kuna tema unistustes töötas robot laitmatult ja lisaks tolmu võtmisele läks ka ise end laadima, tervitas hommikul ukse juures tulevaid töötajaid ja lisaks tõi kohvi ka – ehk klient ootab rohkem kui tegelikult plaanis oli teha või üsna tihti isegi võimatuid ja teostamatuid asju.

Niisugused olukorrad lõppevad üsna tihti tõsiste lahkkelidega kliendi (projekti tellija) ja projekti teostaja vahel. Klient ütleb, et ta ei saa seda, mille eest ta maksab ja mis tema arust oli kokku lepitud. Projekti teostaja mure on olukord mõistlikult lahendada. Kuna nõudeid pole kirja pandud, siis pole teostajal kuidagi võimalik tõestada, et alguses oli vähem kokku lepitud. Kui ta tahab klienti hoida, siis peab ta ilmselt järele andma või tegema rohkem kui plaanis, aga sama eelarve sees. Halvemal juhul klient lihtsalt ütleb, et tema ei maksa selle eest, mida ta pole tellinud ja teostajale jääb kätte poolik projekt ja halb reklaam tema kohta.

2. klient rõõmustab - oi kui tore masin - talle saaks ju veel juurde teha diskotulukesed ja üleüldse oleks ju äge, kui robotile saaks panna külge kaamera, et robot saaks päeval kontoris ringi sõita ja ma saaksin jälgida, kuidas mu töötajad teevad tööd – ehk klient tahab projekti keerukust lisada, aga eeldab, et see käib olemasolevatesse projekti ajaraamidesse ja raha

sisse, sest „see oli ju projekti algusest peale kokku lepitud“. Ja jälle on kaotajaks projekti teostaja.

Seepärast tuleb nõuded kohe projekti alguses fikseerida, korrektselt kirja panna ja kliendiga läbi rääkida, et kõik saaksid ühtmoodi aru ja oleks selgelt piiritletud, mis projekti käigus tehakse, mis mitte ja millise kvaliteediga. Eelkõige on nõuete fikseerimine projekti teostaja mure, sest temal on rohkem kaotada.

Kuidas nõudeid kirja panna?

Nõuete kirjapanekul on oluline, et:

1. nõuded oleksid konkreetsed - kõik saaksid nõuetest ühtmoodi aru ja oleks täpselt aru saada, et projekti tulemusena oodatakse
2. nõuded oleksid mõõdetavad - et oleks võimalik pärast kontrollida, kas nõue on täidetud või mitte

Mõistlik on nõuete kirjapanekul jälgida, et sarnased nõuded oleksid koondatud gruppidesse, näiteks nõuded mehaanikale, nõuded tarkvarale, üldised nõuded jne.

Näited nõuetest

Läheme tagasi oma kontori koristamise roboti juurde ja vaatame üle mõned nõuete näited. Meie klient soovis, et robot koristaks öösel ära kõik kontoriruumid. Kuidas me selle soovi võiksime nõuetena kirja panna?

- robot võiks koristada öösel
- robot peaks ära koristama kogu kontori

Mis on nende nõuete juures valesti? Nõuded ei ole kirja pandud konkreetselt, ega mõõdetavalt.

Vaatame esimest nõuet: „robot võiks koristada öösel“. Kuidas peab robot aru saama, millal on öö? Kas öö tellija jaoks algab kell 17/18 kui töötajad koju lähevad? Või algab öö päikeseloojanguga ja lõpeb päikesetõusuga? Teiseks sõna „võiks“ kasutamine nõudes - kas see tähendab, et kui robot ei korista öösiti või ei korista üldse, siis pole ka midagi valesti?

Seega peaksime nõuet täpsustama. Oletame, et meie klient nõuab, et robot töötaks öhtul kella 21-st hommikul kella 6-ni. Korrektned nõue oleks:

- Robot peab koristama öösiti, igal öösel alates kella 21-st 6-ni

Kas tundub, et nüüd ongi kõik? Tegelikult siit keeruline osa alles algab. Kui nüüd natuke mõelda, peaks projekti teostajal tekkima veel mitu küsimust. Mõned neist:

1. Kui robot peab töötama vahemikus 21-6, siis:
 - a. kas robot peab ise aru saama, millal koristusaeg algab ja lõpeb või on firmas keegi, kes roboti sisse lülitab igal öhtul ja seisma paneb igal hommikul?
 - b. kas robot peab töötama oma energiaallika (aku) pealt või on juhtmega seinas?
 - c. kui robot peab töötama oma energiaallika pealt, siis kas ta peab suutma vastu pidada järjest korraga 9h?

- d. kui robot ei pea vastu pidama korraga 9h, siis kas see peab käima end ise laadimas? Või on mõni firma töötaja, kes roboti akut vahetab? kui on, siis kuidas see töötaja teab, et akut on vaja vahetada?
- e. kui robotil keset ööd tolmukott täis saab, siis kuidas robot käituma peab?
- f. kui robot töötab öösel, kas siis kontoris on tuled ära kustutatud ja robot peab töötama pimedas?

jne ...

Kui ülesande üle laiemalt mõelda, tekib küsimusi juurde:

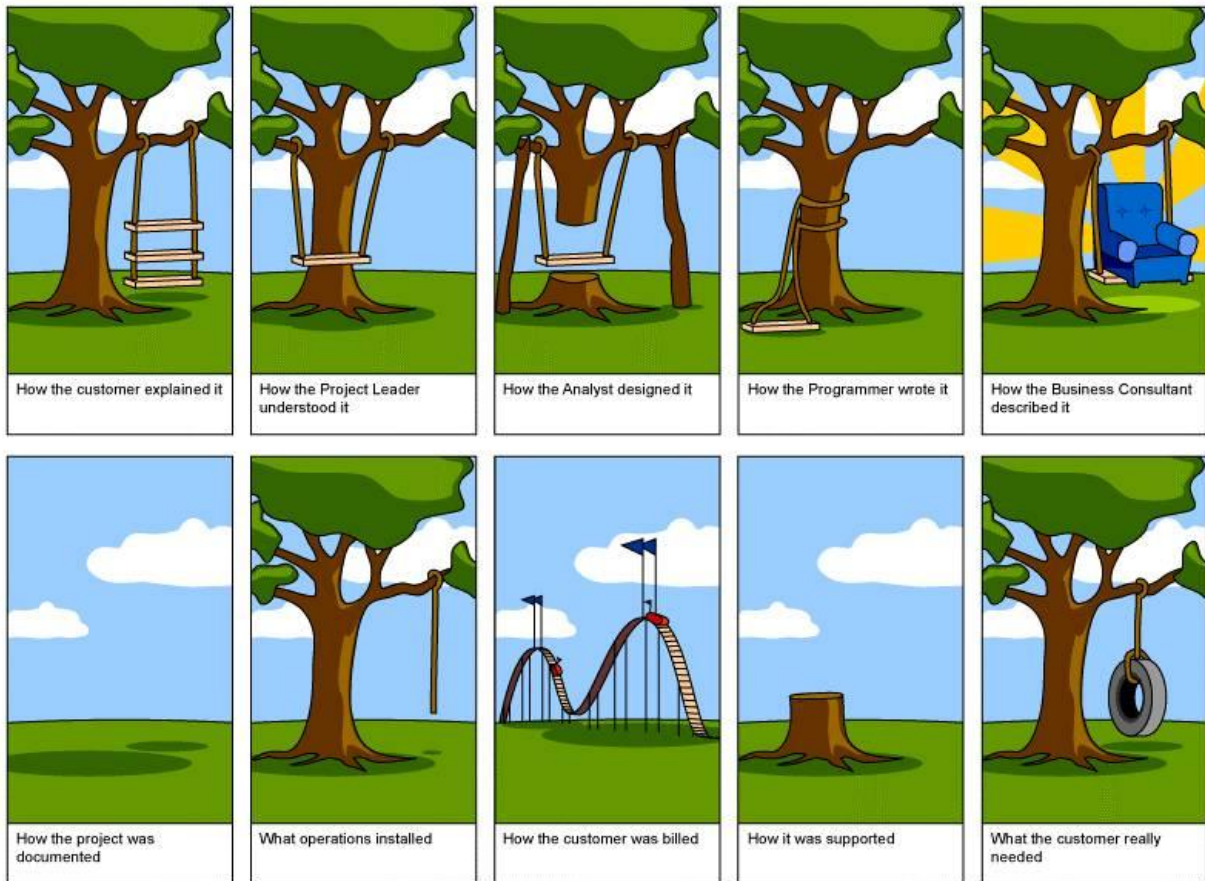
- 2. Kuidas robot teab, millised ruumid ta ära koristama peab?
- 3. Kuidas robot orienteerub ruumides? Kas ruumidesse on võimalik panna majakad, mis piiravad roboti koristamise ala? Kas ruumides on mööbel ja esemed alati sama paigutusega? Kas ruumides on kergesti purunevaid ja ohtlikke esemeid, mida ei tohi ümber ajada? (kujutage ette robotit laboris, kus riiulites on inimesele ohtlikud viirused)
- 4. Kas robot töötab alati samades tingimustes või on ruumides tingimused märgatavalt erinevad (näiteks, kas robot peab suutma ka niisket libedate kiviplaatidega vannituba koristada või sõidab ainult vaiba peal?)
- 5. Kui kontoriruumid asuvad kahel korrusel ja vahel on trepp, kas robot peab suutma siis ka ühelt korruselt teisele liikuda?

jne ...

Kõik need küsimused tuleb kliendilt üle küsida ja vastused fikseerida konkreetsete ja mõõdetavate nõuetena.

Nõuded vs kliendi soovid

Siin on veel üks koht, mille üle tuleb mõelda - klient, kui üldiselt robotikavõõras inimene, hakkab pahatihti seletama oma soove enda mõeldud tehniliste lahendustena, mitte ei sõnasta neid soovidenäidetena ja nõuetena. Legendaarne pilt tarkvaranõuete kohta (©2005 Paragon Innovations, Inc.):



Oletame, et klient arvas, et robot peab suutma ise laadimas käia. Siis tema võib seletada seda nii: "Jah, robotil võiks vist olla küll küljes selline käsi, mis võtab robotist välja juhtme ja paneb selle seina ja kui aku on täis, siis tõmbab see käsi juhtme jälle seinast välja ja kerib kokku ja siis saab robot uuesti tööle minna."

Tundub naljakas? Tegelikult on see üsna tavapärane asi, sest klientidele tunduvad jalad ja robotkäed jms elementaarne osa robotite juures. Tegelikult tuleb siit välja lugeda nõude jaoks, et "robot peab suutma ise leida laadimispunkti, ise end laadida ja kui aku on täis, siis laadimine lõpetada". See, kuidas tehniliselt lahendada, tuleb välja mõelda teostajal ja kirja panna lahenduse dokumentatsiooni ning seletada kliendile, et see on mõistlikum, odavam, töökindlam, ja lihtsam kui robotkäsi.

Klientidega suhtlemine

Esialgu võib tunduda, et suhtlemine klientidega on elementaarne ja sellest ei peakski rääkima. Samas praktika näitab, et paljudes ettevõtetes ei osata klientidega suhelda. Seetõttu pühendame käesoleva materjali nõuannetele teemal: „Kuidas suhelda klientidega?“

Põhilised probleemikohad klientidega suhtlemisel:

- Emotsionaalsus ja ebaviisakus
- Kliendile tagasiside andmine
- Kliendi teadmatus

Emotsionaalsus ja ebaviisakus

Tihti võime kokku puutuda klientidega, kes on ebaviisakad, otsekoheised, kergesti ärrituvad vms. Kõige suurem viga, mida nende klientidega suhtlemisel tehakse, on otsused ja käitumine emotsioonidele tuginedes – vastates samamoodi ebaviisakustega või ärritavate kommentaaridega.

Alati tuleb jääda viisakaks! Ropendamine ja sõimusõnade kasutamine ei tule kõne allagi.

Ebaviisakas käitumine emotsioonide põhjal on otseselt „õli tulle valamine“ – juba ärritunud klient vihastab veelgi ja nii ongi lõpuks tüli valmis. Paremalt juhul võtab suhete lappimine kliendiga väga kaua aega, halvemal juhul klient lihtsalt „võtab oma raha“ ja läheb järgmise teenusepakkuja juurde. Ja kuna ettevõtte eesmärk on teenida kasumit, siis ei aita kliendi välja vihastamine kuidagi sellele eesmärgile lähemale.

Lisaks kliendi kaotamisele võivad kaasned ka muud ebameeldivused – vihane klient levitab meie kohta halba reklaami. Olgugi, et probleemi põhjuseks on lihtsalt emotsioonidele tuginedes ebaviisakas käitumine, meie klient näeb seda teisiti ja räägib meie kohta, et oleme ebakompetentsed, ega saa oma tööga hakkama ja pakume väga halva kvaliteediga teenuseid (näiteks roboteid). Siin ei ole üldse oluline, kui osavad me oma töös tegelikult oleme – selle halva reklaami tõttu ei taha paljud meie juurde tulla ja meil polegi võimalust näidata, et tegelikult suudame projekteerida ja valmistada töökindlaid roboteid.

Näide

Klient tellib meie käest roboti, mis igal hommikul sõidab tema laua juurde ja valab tassi kohvi täis. Oletame, et tegemist on tüüpilise kliendiga, kes tuleb oma tellimusega ja suurest agarusest ootab, et robot on juba järgmisel päeval valmis. Meie seletame talle, et nii kiiresti ei saa ja lepime kokku, et näitame talle kuu aega pärast projekti algust prototüüpi.

Klient on siiski väga kärsitu ja kirjutab meile, et soovib tulla prototüüpi vaatama ja oma kohvitassiga katsetama nädal aega varem – 3 nädalat pärast projekti algust. Seletame kliendile, et see on liiga vara, aga klient ikka käib peale ja tulebki vaatama prototüüpi nädal varem.

Kohale jõudes klient pettub, kuna robot ei suuda isegi tassi üles leida ja seletab meile häälkalt, kuidas tema ei maksa mittetöötava asja eest ja millega me üldse tegeleme, kui robot isegi kohvitassi üles ei leia.

Siis vale käitumine on ärritada ja vastata talle, et ta poleks pidanud siis kohale tulema kui ei meeldi ja ei saa tööd teha kui tema segamas käib jne. See ärritab teda veelgi.

Parem oleks näidata kliendile seda, mis tehtud on, ja ta maha rahustada. Võiks seletada veidi põhjalikumalt, et kohvitassi leidmise funktsionaalsuse tegemine võtab rohkem aega ja projektiplaani järgi see saab valmis alles nädal aega hiljem.

Selliseid olukordi, kus klient meid endast välja viia võib, tuleb ette ilmselt igas projektis. Näiteks:

- klient ei anna meile õigeaegselt vajalikke algandmeid (näiteks kohvitassi värvi, kuju ja suurust), kuid nõuab tulemusi (robot peab kohvitassi üles leidma) sellegipoolest õigeaegselt.
- Klient tahab juba alustatud projekti muudatusi ja lisada keerukust (lisaks kohvi valamisele võiks teed ka pakkuda), kuid tahab, et need muudatused ja lisad mahuksid sama projekti raamidesse (sama summa eest, sama valmimise tähtajaga)
- Jne.

Liigne emotsionaalsus on probleem, sest seda on endal väga raske märgata. Alati, kui tunnete, et klient vihastab teid välja (näiteks kirjutab väga ärritava kirja), siis enne talle vastamist tuleks rahuneda ja oma vastus uuesti üle vaadata (näiteks kliendi kirjale kirjutatud vastus mõne tunni pärast üle lugeda ja proovida rahulikumas toonis kirjutada)

Kliendile tagasiside andmine

Klient üldiselt ei mõtle sellele, et meil võib olla tema projekti kõrval ka teisi projekte teistelt klientidelt. Tema jaoks kehtib mõte „mina maksan, siis ma tahan ka korralikku teenust“. Seega klient ootab, et temaga tegeletakse ja näidatakse huvi üles. Ta ootab meilt kiiret reageerimist ja tagasisidet.

Kliendile on oluline teada, et tema mure või sooviga tegeletakse. Isegi, kui me kohe täpselt ei oska vastata, siis tuleks anda kliendile märku sellest, et oleme tema murest/soovist teadlikud ja tegeleme sellega.

Näide:

Klient kirjutab e-maili küsimusega „ Millal kohviroboti prototüüp on nii valmis, et see oskab minu kontoris sõita minu laua juurde?“

Kui me täpselt ei tea, siis oleks õige kliendile kiiresti vastata midagi sarnast: „Uurime järgi ja anname teada, millal see võimalik on“ ja hiljem, kui me juba teame, anda õige vastus.

Vale klient ootama jätta, kuna me kohe talle vastata ei oska. Tal tekib tunne, et tema projekt ei ole meie jaoks oluline ja seetõttu võib arvata, et teeme oma tööd lohakalt.

Kliendi teadmatus

Enamasti on klientidel olemas visioon ja idee, aga puuduvad täpsed teadmised, kuidas seda ideed teostada. Sellepärast klient meie juurde tulebki – et saada meie ekspertteadmistele tuginedes nõu ja abi. Kuna kliendil endal need ekspertteadmised puuduvad, siis on kliendi visioon enamasti ebarealistlik.

Näide

Läheme tagasi oma kohviroboti näite juurde. Kliendi visioon kohvirobotist on naist meenutav humanoidrobot, kes kannab kõrgete kotsadega kingi ja käes kohvikannu. Meie teame paremini – mõistlikum on jalgade asemel kasutada rattaid jne. Klient seda ei tea.

Vale on kliendi visiooni peale kõva häälega naerma hakata ja kommenteerida: „Iga loll teab ju, et sellised robotid on ainult filmides. Kas te üldse teate ka, palju sellise humanoidi välja töötamine maksab? Teil pole ilmselt rahagi, et sellise asja eest maksta.“ Niisuguse käitumise peale klient solvub ja enam meie juurde tagasi tulla ei taha.

Õige oleks kliendile ilma tema teadmatuse üle naermata seletada, millised lahendused on robotikas kõige lihtsamad, töökindlamad ja enim kasutatud (näiteks rattad). Ja siis selgitada kliendile, milline lahendus oleks mõistlikum kui humanoid.

Meeskonnatöö korraldamine

Üks suuremaid põhjuseid projektide ebaõnnestumisel on meeskonnatöö puudumine või selle halb korraldamine. Käesolevas materjalis vaatame lähemalt:

1. **Kuidas jagada tööd meeskonnas?**
2. **Milliseid vahendeid kasutada meeskonnatöök?**
3. **Mis on meeskonnatöö reeglid ja milleks neid vaja on?**

Tööjaotus meeskonnas

Iga projekti raames on vaja ära lahendada palju ülesandeid erinevatest valdkondadest. Kuidas tagada, et kõik saaks tehtud ja nii nagu vaja? Üks võimalus selleks on jagada kõik vastutus projekti eest meeskonnaliikmete vahel ära – iga meeskonnaliige vastutab ühe või kahe valdkonna eest ja jälgib, et selle valdkonna ülesanded oleksid lahendatud korrektselt ja õigeaegselt ning mõistlikult.

On erinevaid võimalusi valdkondade vastutuse jagamiseks. Mõned näited:

- Jagada ära vastutus tulenevalt tegevusvaldkondadele - üks meeskonnaliige vastutab elektroonika eest, teine mehaanika eest, kolmas madalama taseme programmeerimise eest jne. Selline lahendus sobib hästi siis, kui iga liige on vastavas valdkonnas kõige pädevam (nö ekspert).
- Jagada ära vastutus tulenevalt lahenduse eripäradest – üks meeskonnaliige vastutab roboti liikumismooduli eest, teine ümbruse tajumise mooduli eest (videotöötlus, erinevate andurite kasutamine jms). See lahendus töötab hästi siis, kui meeskonnas enamus liikmeid on pädevad mitmes valdkonnas.

Ei tohi unustada, et lisaks elektroonikale, mehaanikale ja programmeerimisele on vaja meeskonnas teha ka muud – suhelda klientidega, koostada ja jälgida eelarvet, hoolitseda materjalide tellimise eest jms. Selleks peaks olema meeskonnas üks inimene (projektijuht), kes jälgib alati üldist projekti seisuga ja tegeleb küsimustega, mis ei kuulu ülejäänud rollide või vastutusvaldkondade alla.

Vahendid meeskonnatöök

Meeskonnatöö paremaks korraldamiseks tuleb valida välja vahendid, mida töös kasutatakse. Mõned näited:

- Wiki (hea võimalus projekti dokumentatsiooni hoidmiseks) – kõik saavad alati jälgida ja täiendada. Antud kursuse raames on kasutusel Moodle'i wiki.
- Tasklist (aitab jagada ja hallata ja planeerida töid ning jälgida tööde seisuga) – Antud kursuse raames kasutusel TÜ Robotiklubi digiwiki tasklist.
- Suhtlusvahendid nagu Skype ja MSN Messenger – võimaldavad meeskonnal suhelda. Eeliseks on, et isegi kui meeskond pole kogu koosseisuga alati olemas, on võimalik hiljem lugeda, mida teised kirjutasid ja ajaloost varasemaid vestlusi jälgida.
- GoogleDocs – võimalus hoida dokumente ühes kohas, kusjuures kõigil meeskonnaliikmetel on võimalus jälgida ja redigeerida dokumente ning jooksvalt jälgida teiste liikmete poolt tehtud muudatusi.

- Google'i kalender – võimaldab meeskonna jaoks vajalikku tähtaegadega seotud infot hallata, samamoodi meeskonna koosolekute/eemalolekute (reisid vms) aega arvestada ja planeerida meeskonna töös.
- SVN – hea vahend (eelkõige koodi) failide versioonide haldamiseks.

Kindlasti on võimalusi veel ja need ei piirdu ainult ülal loetletud variantidega.

Meeskonnatöö reeglid

Et meeskonna töö oleks efektiivne ja ei tekiks hiljem probleeme, tuleb projekti alguses leppida kokku teatud reeglid, millest kõik meeskonnaliikmed kinni peavad. Mõned näited reeglitest, mis võiksid olla kasulikud:

- Lepitakse kokku töö maht, mida meeskonnaliikmed iga nädal ära teevad. See aitab projekti tegevusi paremini planeerida. Näiteks 10 tundi nädalas iga meeskonnaliikme kohta.
- Lepitakse kokku kindel nädalase koosoleku aeg (see koosolek võib olla näiteks Skypes või ka isiklikult laboris vms). Sellel koosolekul vaadatakse üle, mis eelmisel nädalal on tehtud, millised on lähenevad tähtajad, seatakse eesmärgid ja planeeritakse tööd järgmiseks nädalaks.
- Lepitakse kokku mõned ühised laboris koos töötamise ajad – elu näitab, et tööd edenevad paremini ja kiiremini kui meeskond töötab koos. Probleemid tekivad, kui robotit ehitatakse eraldi ja hiljem alles saadakse kokku ja selgub, et iga eraldi arendatud jupp võttis mitu korda rohkem tööd ja aega kui mõistlik ning lõpuks ei sobi teiste arendatud osadega. Kõiki neid probleeme saab vältida koos laboris töötades.
- Jne.

Tasklisti täitmisest

Igal meeskonnal on kohustuslik kasutada oma töös tasklisti. Seda tuleb täita nii, et:

1. Tasklistist oleks meeskonnatöös kasu (täitmine lihtsalt selleks, et „oleks midagi kirjas“ raiskab roboti ehitamiseks vajalikku aega, ega aita meeskonna tööd korraldada)
2. Tasklist annaks ülevaate projektis tehtud tööst ja projekti hetkeseisust

Kohustuslik on tasklisti täita esmaspäevase praktikumi lõpus, planeerides meeskonnaliikmetele töö ära kogu nädalaks. Tööde jagamisel jälgige, et kõik vajalik saaks tehtud, kõigil meeskonnaliikmetel oleks tööd ja keegi poleks tööga üle koormatud.

Tasklisti järelvalve on projektijuhi ülesanne ja vastutus.

Uue taski tegemine

Uute taskide lisamine tasklisti peaks olema eelkõige projektijuhi ülesanne (see muidugi ei tähenda, et ülejäänud meeskonnaliikmed ei või töid tasklisti lisada). Mõistlik on rääkida kõik tööd läbi meeskonnaga või vähemalt inimesega, kes seda tegema hakkab, et ei oleks hiljem probleeme sellega, et tasklisti kirjutatakse võimatu töö või antakse sellele täiesti ebareaalne hinnang.

Uue taski tegemisel tuleb täita ära väljad:

- **Title** – kokkuvõtlik pealkiri tööle. See peaks lühidalt kokku võtma töö sisu nii, et tasklisti koondnimekirja vaadates oleks võimalik aimata, mida selle töö raames tehakse.
Näiteks: Halb taski pealkiri on: programmeerimine – see ei anna mitte mingit infot selle kohta, mida programmeeritakse (on see videotöötlus, mootorite juhtimine vms?). Parem oleks näiteks: Triblamismehhanismi mootori juhtimise programmi kirjutamine.
- **Summary** – töö detailne kirjeldus. Siia tuleb kirjutada lahti töö sisu piisava detailsusega – see tähendab, et kõik meeskonnaliikmed peavad olema võimelised aru saama selle töö sisust. Samuti tuleb siia kirjutada ka töö lõpuks oodatav tulemus.
Näiteks – mehaanika detailide valmistamise korral detailide nimetused ja kogused, viited joonistele jms, dokumenteerimise taski korral tekitatavate dokumentide nimetused-kirjeldused, testimise taski korral planeeritud testid.
- **Progress** – töö progressi kirjeldus. Siia tuleb kirja panna see, mis töö käigus valmis sai (näiteks testimise taski korral kokkuvõtte testi tulemustest, elektroonika tellimise taski korral tellitud komponentide nimekiri ja kogused, tarneajad jms) või kui töö käigus esines probleeme, mis muutsid esialgset ülesandepüstitust, siis ka see tuleb kirja panna. Samamoodi tuleks progressi alla märkida ka põhjendused tähtaegade edasi lükkamise kohta.
- **Type** – taski tüüp. Valida on:
 - **Mechanics** – mehaanika tööd (disanimisest ehitamiseni välja)
 - **Electronics** – elektroonika tööd
 - **Programming** – programmeerimise tööd
 - **Doc** – projektiga seotud dokumenteerimistööd
 - **Bug** – igasugused veaparandused (seda ilmselt projektis väga palju vaja ei lähe). Seda tüüpi on hea kasutada siis, kui roboti juures tuleb välja mõni viga, mida kohe parandada ei saa, aga selleks, et ära ei ununeks, on see mõistlik taskina kirja panna

- **Test** – roboti testimised.
- **Other** – igasugused muud tööd, mis ülaltoodud jaotusesse otseselt ei sobi (näiteks igasuguste tellimuste tegemine, kauplustest roboti juppide ostmine, tasklisti täitmine, sponsoritele kirjutamine jms)
- **Status** – töö olek. Valida on:
 - **New** – uus töö
 - **Assigned** – töö on kellelegi määratud
 - **Resolved** – töö on ära tehtud
 - **Closed** – task on kinni pandud (näiteks kui selgub, et pole vaja teha või ei saa teha vms), sisuliselt on closed seisus taskid, mis on tegemata ja mida pole vaja teha ka.
- **Priority** – töö prioriteet. Valida on:
 - **Low** – kasutada siis, kui kunagi on vaja töö ära teha, aga tähtaeg pole oluline ja valmis saamisega pole kiire
 - **Normal** – peaks olema tavaline taski prioriteet
 - **High**
 - **Top** – see peaks olema ka üsna harva esinev ja väga kriitiliste taskide jaoks
- **Assignee** – taski teostaja. Tasklistis peaksid kõik taskid olema projekti osalistele määratud. Unassigned seisu võib olla töö ainult siis, kui on vaja hetkeks mingi task tekitada, et meelest ära ei läheks ja ei ole veel teada teostajat. Projektijuhi kohus on hiljemalt paari päeva jooksul pärast taski tekitamist leida taskile täitja ja Unassigned asemele määrata teostaja.
- **Due** – töö valmimise tähtaeg
- **Estimate** – töö tegemiseks kuluva aja hinnang (tähele peaks panema, et tähtaeg ja hinnang ei ole samad asjad. **Näiteks:** task – sponsorile kirja kirjutamine. Hinnang (estimate) võib olla näiteks 4 h, samas tähtaeg (due) võib olla taski tekitamisest näiteks 1 nädal hiljem). Hinnanguid tuleks kirjutada tundides, näiteks hinnang 1 päev on halb sellepärast, et pole teada, kas 1 päev on 8 h, 24 h või 2 h mingil õhtul.
- **Spent** – tööle tegelikult kulunud aeg (ei pea ühtima hinnanguga). Seda on hea vaadata, et näha, kui hästi osatakse hinnata tööle kuluvat aega ja hiljem, uute tööde korral seda arvesse võtta. Kui inimesed ei ole varem hinnanguid töödele andnud, siis on üsna tavaline, et tegelikult kulub aega rohkem kui esialgu arvatakse.
- **Notify e-mail** – märkides siia linnukese, siis saab saata teavitusmeili sellele, kelle nimele task suunatakse
- **Submit** – salvestab taski ja suunab tagasi tasklisti

Taskide tegemisel tuleb tähelepanu pöörata sellele, et:

- Task ei oleks pikem kui maksimaalselt 1 nädal. Kui töö, mida kirja tahate panna, on pikem kui 1 nädal, siis jagage see mitmeks alamtaskiks. Miks? Sest:
 - mida suurema mahuga töö ette võetakse, seda ebatäpsemad on hinnangud – ja nii on suurem tõenäosus, et ette võetud töid ei saada õigeaks ajaks valmis.
 - Pika tähtaja korral tekib kiusatus töö alustamisega venitada (kui töö jaoks on aega 3 päeva, siis tegutsema hakatakse alles viimase päeva hommikul/õhtul). Niimoodi kaotab meeskond projekti tegemiseks mõeldud aega ja seega ka tulemus ei ole niisugune nagu oodatakse.

- Taski kirjelduses peab olema kirjas oodatav tulemus. Miks? Sest muidu tekib kiusatus kirjutada töid, millega vormiliselt näidata, et midagi tehakse, aga mis realselt projektis progressi ei tekita.

Näiteks: Kui töö sisuks on mingi teema uurimine/lugemine jms, siis peab olem oodatav tulemus midagi, mida saab mõõta ja millest on ka ülejäänud meeskonnal kasu – näiteks ettekanne uuritud teema kohta meeskonnale või mõned leheküljed kokkuvõtet selle kohta, mida sellest uurimisest teada saadi.

Taski uuendamine

Tasklisti haldamine ei piirdu ainult uute taskide tegemisega. Taskide seis peab uuendama. Uuendamiseks ja täpsustamiseks on taski kirjelduses lahter **Progress**. Kindlasti tuleb kirjutada kommentaar taski tähtaja muutmisel – miks tähtaega edasi lükati. Taski uuendamisel lisada kommentaarile ka kuupäev.

Taski uuendamisel tuleks märkida ka jooksvalt juba tööle kulunud aeg (**Spent**).

Taski uuendamisel on võimalik saata teavituseil taski teinud inimesele. Kui tööde lisamine tasklisti võiks olla eelkõige projektijuhi ülesanne, siis taskide uuendamine on tööde teostaja ülesanne ja projektijuhi kohustus on seda jälgida ja vajadusel juhtida tähelepanu unarusse jäänud taskidele.

Taski kinni panemine

Taski sulgemisel tuleb kindlasti ära märkida, kui palju aega tegelikult taski tegemiseks läks (**Spent**). Task pannakse kinni alles siis, kui kõik taskis märgitud asjad on tehtud ja töö on valmis! Kinni ei tohi panna taski selgitusega „põhimõtteliselt valmis, küll ma pärast vaatan selle viimase asja ka ära“ – see viimane asi tihti ununeb või segab järgmiste tööde planeerimist ja tagajärgi kahetsete te seda Robotexil kui kõik valmis ei ole.

Tasklisti haldamine

1. Tasklisti järelevalve on projektijuhi kohustus. Projektijuht peab jälgima, et:
 - 1.1. Tööd oleks tasklistis kirjas
 - 1.2. Kõigil projekti liikmetel on alati midagi käsil
 - 1.3. Punaseid (üle tähtaja läinud) taske ei tohiks ideaaljuhul üldse tasklistis olla. Kui neid on, siis on projektijuhi kohus uurida järgi miks need on punased ja uuendada ära task – kui lükatakse edasi tähtaega (due date), siis tuleb kirjutada progressi kirjeldusse kommentaar, miks edasi lükati (progress). Edasi lükkamist ei tohiks teha kergekäeliselt – Robotexi edasi lükata ei saa ja iga edasi lükkamine tähendab, et seda, et midagi jääb hiljem ajapuudusest tegemata.
2. Iga meeskonnaliikme kohus on oma nimel olevaid taske uuendada, et alati oleks kõigil ülejäänud meeskonnaliikmetel võimalik jälgida, mis seisus tööd on.

Projekti lahenduse kavandi koostamine

Projekti lahenduse kavand

Projekti nõuetele tuginedes koostatakse projekti lahenduse kavand. Lahenduse kavandi koostamisel tuleb silmas pidada, et lahendus:

- rahuldaks projekti nõudeid
- oleks võimalikult lihtne
- oleks võimalikult odav
- oleks teostatav olemasolevate ressurssidega (rahaliste vahenditega, meeskonna tööga ja ajaliste piiride sees)
- oleks töökindel

Lahenduse kavand sisaldab järgmisi osi:

- lahenduse üldkirjeldus
- lahenduse mehaanika kirjeldus (mehaanika ehituse üldkirjeldus, mehaanika ülesehituse sketš ja joonised, mehaanika koostediagramm, vajalike detailide loetelu)
- lahenduse elektroonika kirjeldus (elektroonika ülesehituse üldkirjeldus, elektroonika seoste diagramm, vajalike komponentide loetelu)
- lahenduse programmi kirjeldus (programmi ülesehituse kirjeldus, jaotus kõrgema- ja madalama taseme loogikaks ja mooduliteks, moodulite kirjeldus, programmi seoste diagramm)
- lahenduse valmistamise eelarve

Lahenduse üldkirjeldus

Lahenduse üldkirjelduses tuleb lühidalt selgitada, mis on projekti eesmärk. Robotika projekti puhul seletada ära, milline robot üldjoontes on, mida ta suudab teha ja milliseid tehnoloogilisi lahendusi kavatsetakse kasutada ning miks. Lahenduse üldkirjeldus peab olema võimalikult üldine, samas piisavalt täpsustatud, et projektiga mitte seotud isikud seda lugedes saaksid ülevaate, mida projekt endast kujutab.

Lahenduse üldkirjelduse koostamisel on abiks projekti jaoks koostatud nõuded. Iga nõude kohta tuleks kirja panna, kuidas roboti juures seda nõuet rahuldada. Vaatame näiteks valverobotit:

- Olgu nõudeks „*Robot peab suutma liikuda ühest ruumist teise*“. Lahenduses kirjutame „*Robot kasutab liikumiseks rattaid*“.
- Olgu nõudeks „*Robot ei tohi otsa sõita teda ümbritsevatele inimestele ja esemetele*“. Lahenduses kirjutame „*Robot kasutab oma ümbruse tajumiseks videokaamerat, millest ta tuvastab teda ümbritsevad objekte*“.
- Olgu nõudeks „*Robot peab kasutama oma energiaallikat*“. Lahenduses kirjutame „*Robot kasutab energiaallikana 2 LiPo akut, millest üks on mootorite liigutamiseks ja teine roboti loogika toiteks*“.
- Jne..

Nii tuleb üle tuleb vaadata kõik nõuded ja igale nõudele kirjutada juurde võimalik lahendus. Kui lahenduseks on mitu mõistlikku alternatiivi, tuleks need kõik kirja panna. Lisaks valitud lahendusvõimalustele tuleb iga nõude juurde kirjutada ka lahenduse realiseerimiseks vajalikud ressursid. Jälgida tuleb, et valitud lahendused oleksid võimalikult lihtsad!

Ülevaatlikkuse mõttes on hea see kõik kirja panna tabelina. Aluseks võiks võtta nõuete tabeli ja lisada sinna juurde veerud: „võimalikud lahendusvariandid“, „vajalikud ressursid (materjalid)“ ja „vajalikud ressursid (oskused, teadmised)“.

Kui lahenduse üldkirjeldus on paigas, vaadatakse eraldi täpsustavalt mehaanikat, elektroonikat ja programmi osa.

Lahenduse mehaanika kirjeldus

Lahenduse mehaanika kirjeldus koosneb:

- mehaanika ehituse üldkirjeldusest
- mehaanika ehituse joonistest ja sketšidest
- mehaanika koostediagrammist
- vajalike detailide loetelust

Mehaanika lahenduse koostamist tuleks alustada nõuete üle vaatamisest. Kui nõuete üldkirjeldus on olemas ja iga nõude juures on olemas võimalik lahendus, siis tuleb nendest välja filtreerida mehaanika kohta käivad lahendused ning need süstematiseerida – saame **mehaanika üldkirjelduse**.

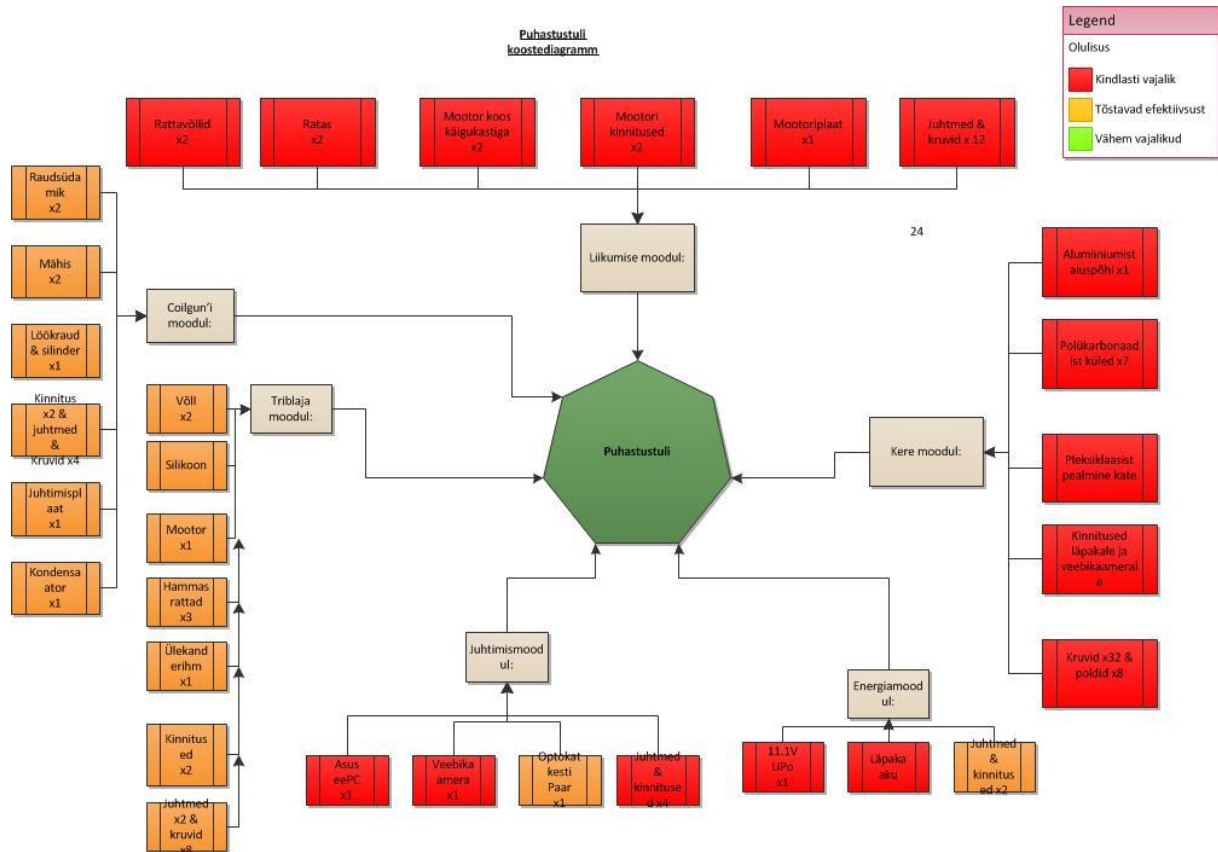
Edasi tuleks mõelda, kuidas mehaanika üldkirjelduse alusel koostada valmis mehaanika – abiks paberile või arvutisse joonistatud **sketšid** (paber ja pliiaats on enamasti kõige kiirem ja lihtsam võimalus esialgse kavandi koostamiseks). Edasi võib kasutada juba paremaid vahendeid – joonestusprogrammid Solid Edge või Solidworks.

Kui üldpilt on selge – on teada, mida tahetakse teha (sõitmismoodul, uste avamise moodul, ...), siis tuleks see lahendus visualiseerida. Üks võimalus selleks on **mehaanika koostediagramm**. See kujutab endast moodulite kaupa kirjeldust sellest, mida üks või teine moodul sisaldab ja kuidas moodulid omavahel seotud on.

Näitena vaatame mehaanika koostediagrammi jalgpalli mängiva roboti „Puhastustuli“ jaoks. Selles on näha, et robotil Puhastustuli on 6 mehaanika moodulit:

- Liikumise moodul
- Kere moodul
- Energiamoodul
- Juhtimismoodul
- Triblaja moodul
- Coilgun'i moodul

Edasi on iga mooduli juures näidatud ära füüsilised detailid, mis sellesse moodulisse kuuluvad. Samuti on joonisel märgistatud ära, mis on kindlasti vajalik, et robot täidaks minimaalselt projekti nõudmisi.

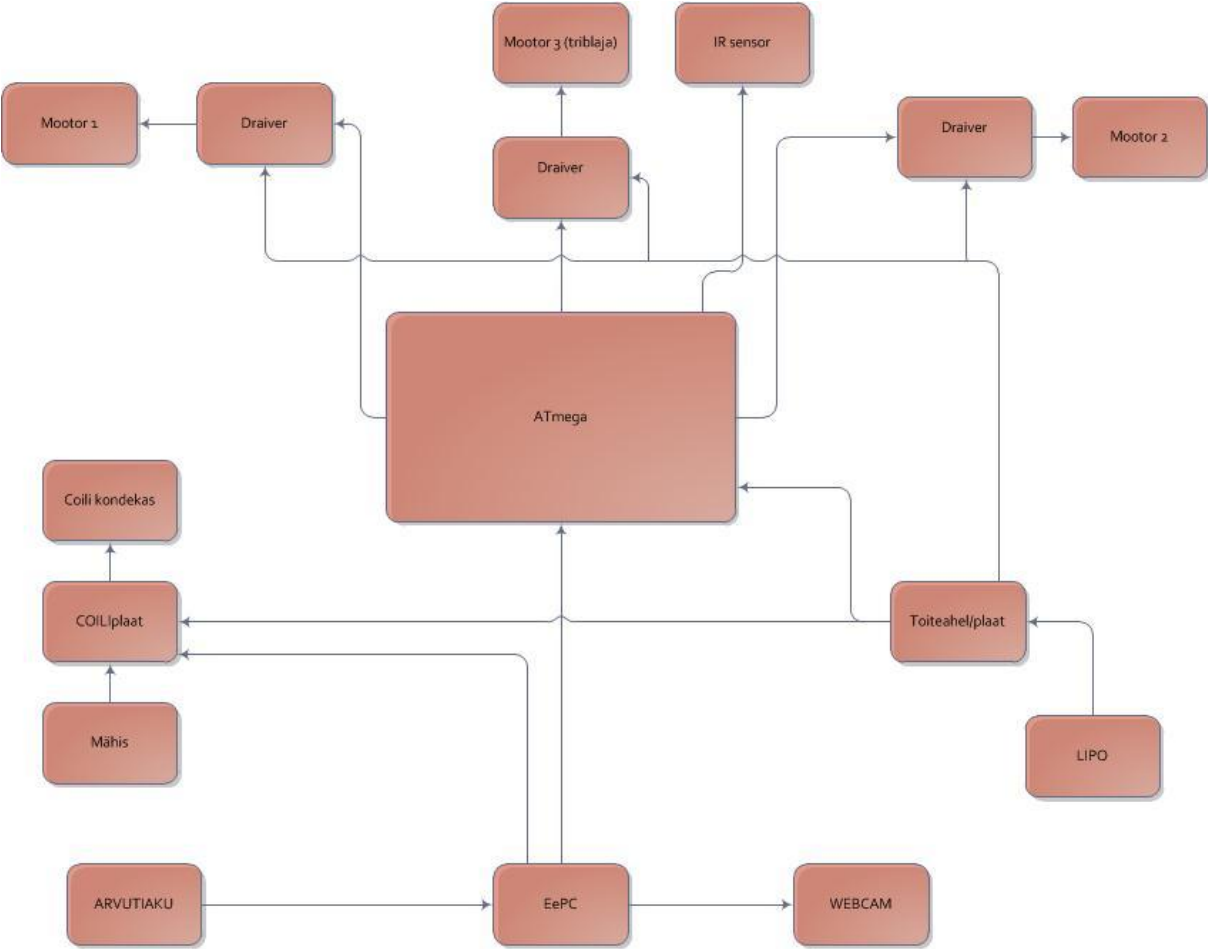


Lahenduse elektroonika kirjeldus

Nii nagu mehaanikas, tuleks ka elektroonikas alustada elektroonika kohta käivate nõuete välja filtreerimisest ning nende süstematiseerimisest – saame **elektroonika üldkirjelduse**.

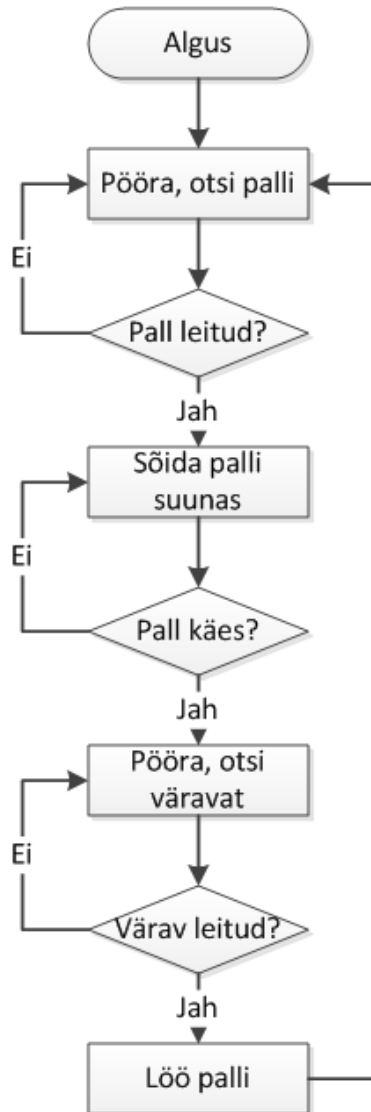
Mõelda tuleks samamoodi moodulitena ja visualiseerida **elektroonika seoste diagrammina**. Näiteks vaatame roboti Puhastustuli elektroonika seoste diagrammi, kus on ära selgitatud roboti elektroonika põhimõtte.

Diagrammilt näeme, milliseid elektroonika osasid lahenduses kasutatakse ja kuidas need on omavahel seotud.



Lahenduse programmi kirjeldus

Lahenduse programmi kirjeldamiseks tuleb alustada sellest, et filtreeritakse välja nõuded, mis on seotud tarkvaraga ja need süstematiseeritakse. Selle süsteemi alusel tuleks paika panna üldine käitumisloogika. Hea ülevaate annab, kui see käitumisloogika vormistada diagrammina, millelt on näha, kuidas robot peaks käituma. Näiteks toome ühe väga lihtsa algoritmiga roboti käitumisdiagrammi:



Edasi tuleks iga tegevuse kohta mõelda, kuidas seda teostada. Mõistlik on jagada programm kõrgema ja madalama taseme loogikaks. Madalama taseme loogika on justkui keha, mis täidab lihtsaid käske – liikumine ja andmete vastuvõtmine ümbrusest. Kõrgema taseme loogika on nõ aju, mis tegeleb mõtlemise ja otsustamisega – kuhu liikuda, mida edasi teha, milliseid andmeid kasutada jne.

Projekti lahenduse analüüsimine

Kui projekti kavand on paigas, tuleb seda analüüsida, leida üles lahenduse riskid ja üritada neid maandada.

Põhilised lahenduse analüüsi osad:

1. **Lahenduse vastavus nõuetele.** Tuleb välja selgitada, kas valitud lahendus rahuldab ära kõik projekti olulised nõuded? Kui suur osa olulisi nõudeid jääb lahendamata, siis valitud lahendus ei sobi ja nõuab muutmist
2. **Lahenduse teostatavus.** Lahendust tuleb uurida teostatavuse seisukohalt. Põhiküsimus on, et kas meeskond olemasolevate ressursidega suudab selle lahenduse projekti mõttes mõistliku ajaga ellu viia? Kui ei, siis tuleb lahendusest võtta välja osad, mis on väga rasked või lausa võimatud teostada ja asendada need teostatavate lahendustega.
3. **Lahenduse keerukus.** Kas valitud lahendus on lihtne või on võimalik mõnd lahenduse osa sama tulemuse saamiseks realiseerida lihtsamalt? Kui jah, tuleks kasutada lihtsamat lahenduse alternatiivi. Mida lihtsam projekt – seda vähem probleeme tekib selle ellu viimisel.
4. **Lahenduse modulaarsus.** Kas lahendus on ehitatud üles nii, et oleks võimalik erinevaid osasid paralleelselt realiseerida? Kui ei, tuleb lahendust ringi teha nii, et oleks võimalik korraga arendada mitut moodulit – kogu meeskonda efektiivselt rakendada.
5. **Lahenduse testimine.** Mõelda, kuidas saaks lahendust testida jooksvalt, töö käigus ja hiljem kui projekti moodulid või kogu projekt on valmis. Kui selgub, et testimine pole võimalik enne projekti lõppu, tuleks lahendust ringi mõelda – oluline on saada töökindel lahendus. See aga eeldab, et lahenduse osasid saab testida juba enne terviklahenduse kokkupanekut.
6. **Lahenduse kulukus.** Kui palju selle lahenduse realiseerimine maksma läheb? Kui mõni komponent või moodul on liiga kallis, siis tuleks otsida odavamaid alternatiive.

Riskide leidmine ja maandamine

Kui põhiline analüüs on tehtud, tuleks välja selgitada projektiga seotud riskid. Riskide allikad on erinevad. Robotika projekti puhul tulevad riskid põhiliselt kolmest allikast:

1. **Riskid meeskonnast** – Näiteks: meeskond ei ole projekti alguses piisavalt pädev projekti ülesannet lahendama. Et seda riski maandada, tuleks leida projekti lahendusest osad, mille elluviimiseks puudub pädevus, ning seejärel võimalusel valida alternatiiv, mille teostamiseks meeskond on pädev või leida võimalused, kuidas vajalik teadmisteressurss meeskonda hankida.
2. **Riskid lahendusest** – Näiteks: lahenduses tahetakse kasutada varem järgi proovimata võimalusi (uued elektroonikakomponendid, varem kasutamata mehaanika lahendused). Sellisel juhul on oht, et lahendus reaalsuses ei tööta. Siin peaks välja mõtlema alternatiivid – mida teha, kui esialgselt valitud lahendus tegelikult ei tööta nii nagu vaja. Oluline on, et sellesse punkti jõudes ei jääks ehitus seisma ja projekt katki, vaid oleks täpselt teada, kuidas edasi toimida.
3. **Riskid tarnijatest** – Tihti ei osata projektide juures hinnata riske, mis tulevad tarnijatest. Oluline on teada, kui kaua aega võib kuluda selleks, et komponent pärast tellimist kohale

jõuaks. Mõne komponendi puhul võib see olla 2-3 kuud, mõne puhul isegi rohkem kui pool aastat. Kui lahendus jääb sõltuma mõnest sellisest detailist, siis tuleks see risk maandada – leida alternatiiv, mis on paremini kättesaadav (jõuab kohale näiteks nädalaga).

Eelarve koostamine

Eelarve koostamisel on esimeseks abimaterjaliks lahenduse kirjeldus ja selle jaoks koostatud diagrammid. Nende alusel tuleb koostada nimekiri lahenduse jaoks vajalikest detailidest ja kogustest.

Kui on olemas detailide loetelu, siis tuleks hakata välja otsima konkreetseid tooteid ja tootjaid/tarnijaid. Alustada võiks otsimist järgmisest nimekirjast:

1. Tööstusplast - <http://www.tplast.ee/>
2. Oomipood - <http://www.oomipood.ee/>
3. Farnell - <http://ee.farnell.com>
4. RS - <http://ee.rsdelivers.com/>
5. YE International - <http://www.yeint.ee>
6. TME - <http://www.tme.eu/en/>
7. Mouser Electronics - <http://se.mouser.com/Home.aspx>
8. IT Metall (Vitamiini tänav Tartus) - <http://www.itmetall.ee/>
9. Schmolz-Bickenbach - <http://www.schmolz-bickenbach.ee/>
10. Kane metall - <http://www.kanemetall.ee/>
11. Proplastik - <http://www.proplastik.ee/>
12. Tolmets - <http://www.tolmets.ee/>
13. Bauhof - <http://www.bauhof.ee/>
14. K-Rauta - <http://www.rautakesko.ee/est>
15. Baltic Bolt - <http://www.balticbolt.ee/toode.php?show=products&parentID=1>
16. Elme metall - http://www.elmemetall.ee/index.php?lang=eng&main_id=519
17. Levadia - <http://www.levadia.ee/>
18. Anaheim Automation - <http://www.anaheimautomation.com/>
19. The Robot MarketPlace - <http://www.robotcombat.com/store.html>
20. SuperDroid Robots - <http://www.superdroidrobots.com/>
21. Fingertech Robotics - <http://www.fingertechrobotics.com/>

Kui täpsed tooted ja tootjad on välja valitud, siis eelarve tuleks koostada tabelina, mille veergudes on detailide kohta kirjas:

- Nimetus (elektroonika komponendid, materjalid detailide valmistamiseks, teenused – laserlõikus, labori rentimine, ...)
- Kogus – eraldi võiks välja tuua, kui palju on kindlasti vaja ja kui palju on tagavaraks
- Hind
- Tarnija (kust ostetakse või kelle käest tellitakse)
- Tarneaeg (kaua läheb et tellitud detailid/materjalid/komponendid) kohale jõuaks

Eelarve koostamisel ei tohi unustada, et eelarvesse tuleb sisse arvestada alati varu. Teada on, et elektroonikas mõnikord komponendid põlevad läbi ning mehaanikas kõik detailid alati ei sobi sellisena nagu esialgselt kavandatud ja tuleb valmistada uued.

Ülevaatlik projektiplaani ja selle koostamine

Projektiplaani koostamine on väga oluline etapp projekti jooksul. Õigesti koostatud plaan aitab tagada projekti õnnestumist. Halvasti koostatud plaan seevastu aitab kaasa ebaõnnestumisele.

Projektiplaani koostamisel tuleb aluseks võtta valitud lahendus, meeskonnatöö reeglid ja teadmised projekti elutsüklist.

Projektiplaani koostamist alustatakse ajaliste piirangute määratlemisest – teada on lõpptähtaeg, millal peab projekt olema valminud. Antud aine raames on selleks lõpptähtajaks kursuse lõpp (esitluse ja aruande esitamise tähtaeg). Kogu projekti tegevused peavad mahtuma ära nendesse ajalistesse piiridesse.

Edasi vaatame valitud lahendust ja meeskonnatöö reegleid. Meeskonnatöö reeglitest saame kätte viisi tööde paigutamiseks projektiplaani (Näiteks kui meeskond leppis kokku, et kõik liikmed teevad nädalas 10 h tööd, siis iga tööpäevaga tuleks tehtud saada 2 h tööd). Valitud lahenduse annab meile tööd, mida on vaja teha, et projekt valmiks. Eraldi vaadatakse mehaanikat, elektroonikat ja programmi ning muid projektiga seotud tegevusi. Igas grupis vaadatakse tegevusi moodulite kaupa (aluseks lahenduste kirjelduste juures koostatud diagrammid ja tabelid).

Tööd paigutatakse projektiplaani ajalises ja tähtsuse järjekorras – kriitilisemad ja olulisemad tööd ette ja viimaseks need, mille tegemata jäämine ei kahjusta oluliselt lõpptulemust. Miks? Sest nii saab projekti üldeesmärki täidetud ka siis, kui kõik tööd ei jõua päris valmis.

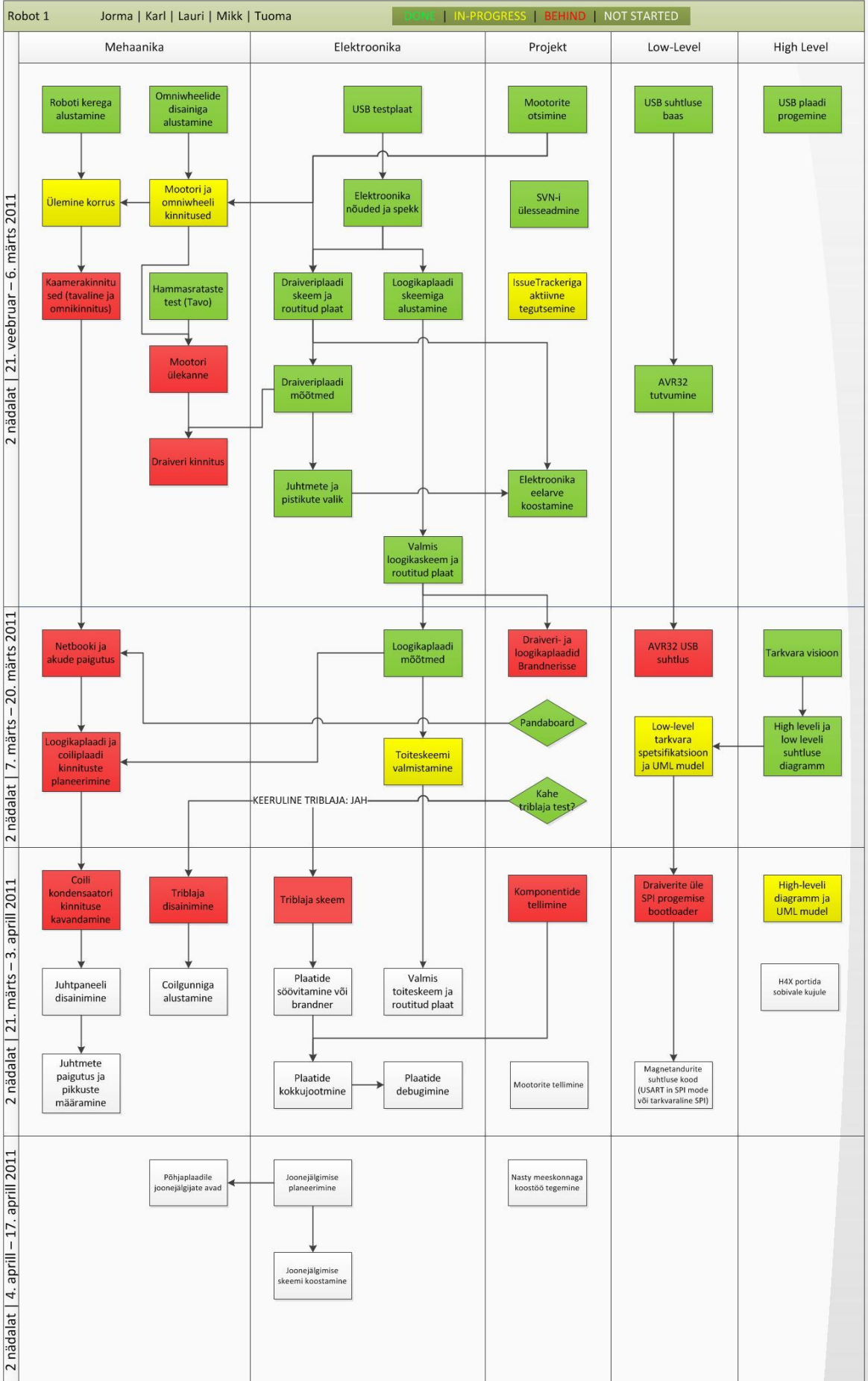
Projektiplaani tuleb uuendada jooksvalt – hea on selleks kasutada värvidega märgistust (näiteks rohelised on tehtud tööd, kollased hetkel käsil olevad tööd jms).

Mida projektiplaani koostamisel tähele panna:

1. Kõikides valdkondades (elektroonika, mehaanika, tarkvara) peaksid tööd toimuma paralleelselt. Halb on näiteks plaan, kus elektroonika tööd algavad pärast seda, kui mehaanika tööd on valminud, tarkvara tööd pärast elektroonikat jne, sest suurendab projekti ebaõnnestumise tõenäosust.
2. Kõik tööd peaksid olema jaotatud meeskonnas võrdselt – ei tohi tekkida olukorda, kus ühele inimesele on planeeritud palju rohkem koormust kui teistele, sest suurendab projekti ebaõnnestumise tõenäosust.
3. Tööd peavad olema prioriteetide mõttes õiges järjekorras – olulisemad tööd eespool, vähemolulisemad hiljem.

Näitena vaatame projektiplaani, mis on toodud järgmisel lehel.

Kindlasti tuleb projektiplaani jätta sisse varu (mõnekuuse projekti korral mõned nädalad), sest töö käigus tekib probleeme, mida projekti alguses ei suudeta ette näha ja enamasti hinnatakse üle võimeid ja alahinnatakse töömahtu. Varu jätmine aitab maandada riski ebaõnnestuda.

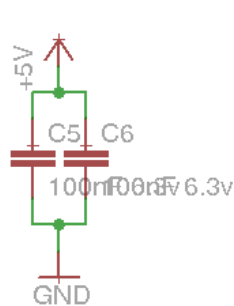


Elektriskeemide vormistamine (autor: Joel Kuusk)

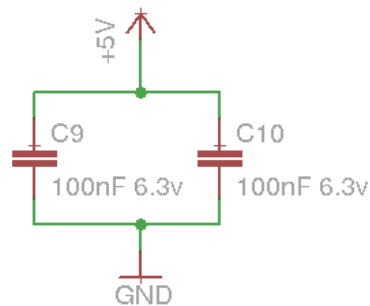
Elektriskeem peab sisaldama kogu vajalikku infot skeemi kokku panekuks. See peab olema arusaadav ka antud projektiga mitte kursis olevale inimesele, eeldusel, et ta siiski elektroonikaskeemidest üleüldiselt aru saab. Skeemi ainus mõte ei ole selles, et Eagle sul trükkplaadi disainimisel signaalid ise komponentide vahel ära ühendaks. Skeemi on vaja ka näiteks komponentide plaadile jootmiseks (seda ei pruugi sa ju alati ise teha), veaotsinguks, teiste inimestega skeemiga seotud küsimuste arutamiseks jne. Skeemi korralik vormistus muudab seda kõike oluliselt lihtsamaks.

Mõned konkreetset näpunäited elektriskeemide joonistamiseks

Ära kirjuta kirjasid üksteise otsa

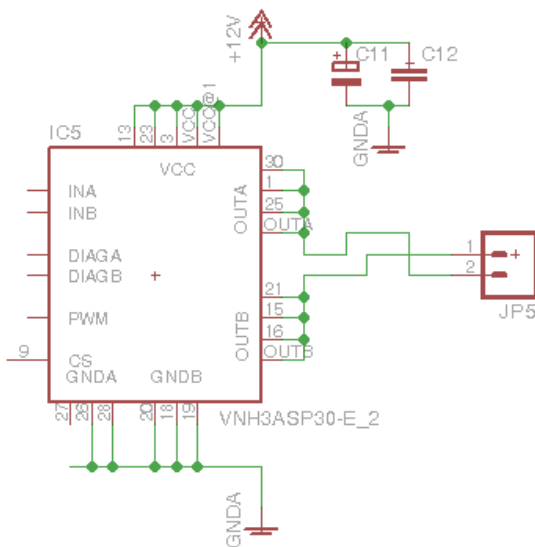


VALE

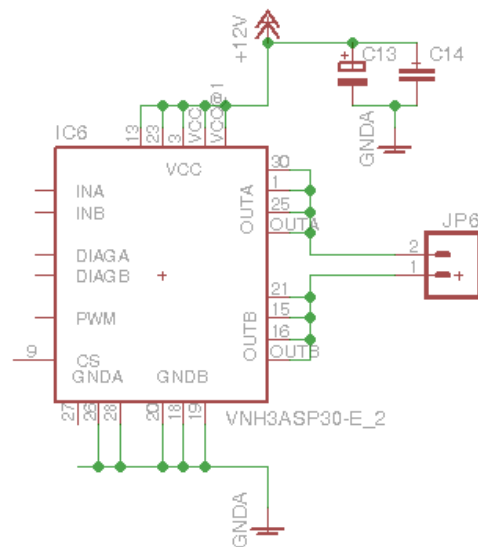


ÕIGE

Ära vea põhjusteta juhtmeid risti ega tee mingeid imelikke haake

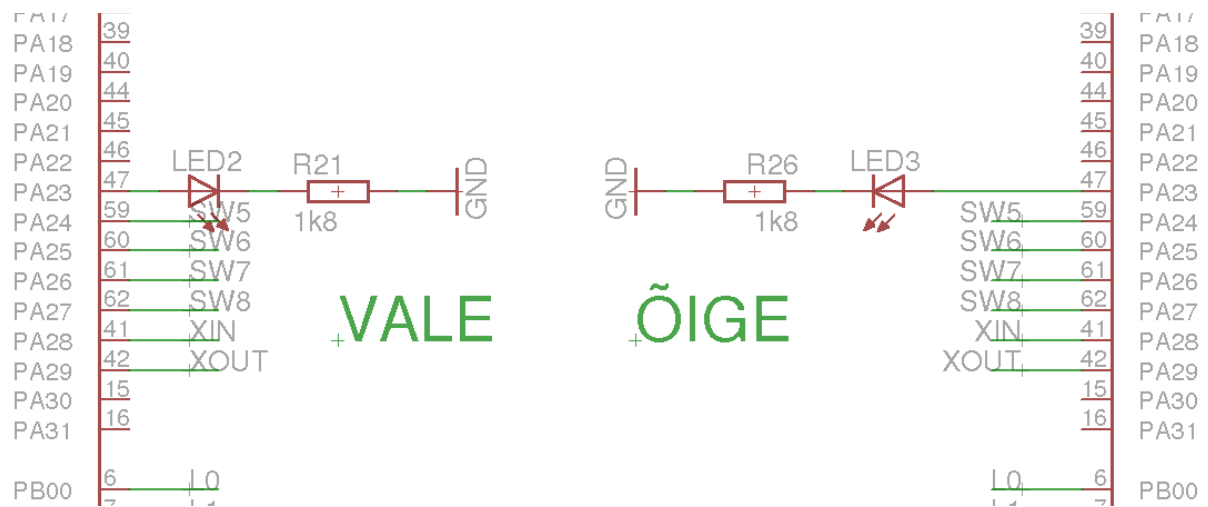


VALE

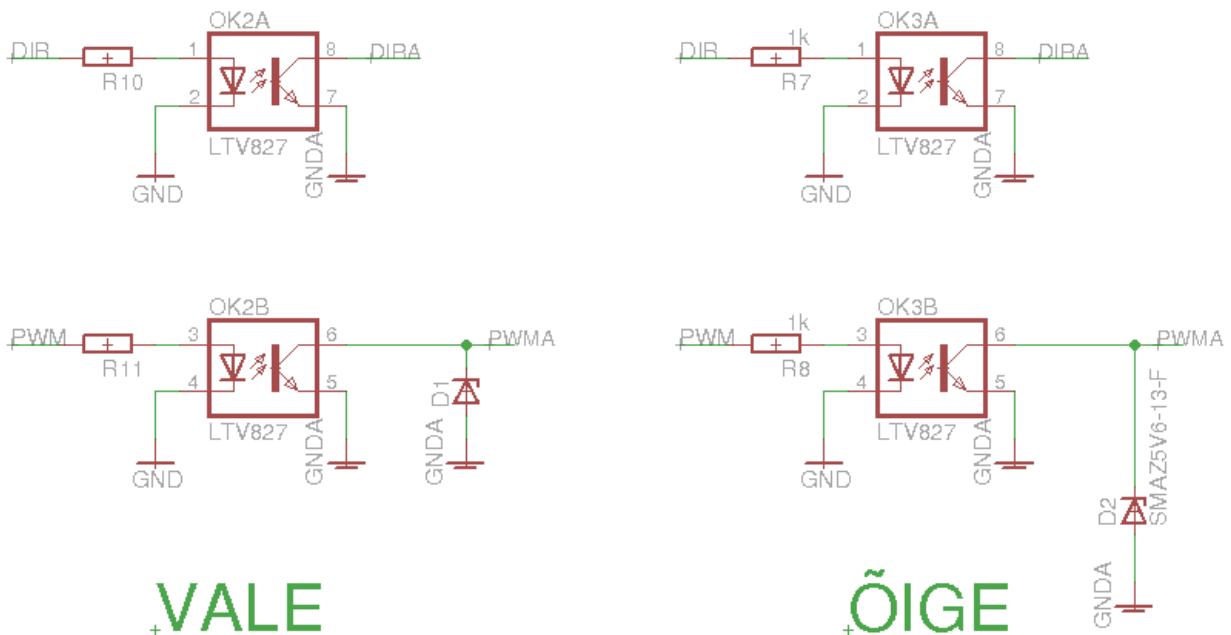


ÕIGE

Ära joonista komponente muude asjadega otsakuti

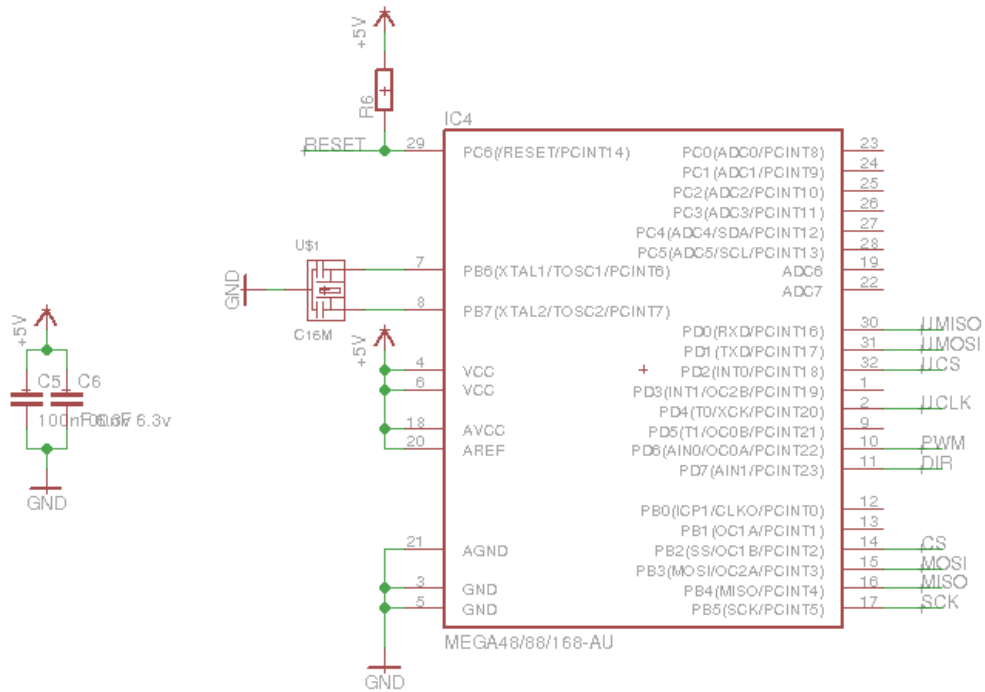


Ära jäta skeemile nime või nominaalita komponente. Isegi kui sa ei tea näiteks, milline takisti võiks sobida ja sa kavatsed selle katse-eksituse meetodil kindlaks teha, siis millestki peate ju ikkagi proovimist alustama. Pange vähemalt seegi kirja. Ära karda kirja panna ka väga pikki nimesid, kui mõnel komponendil peaks selline olema. Kui sa kirjutad stabilitrone juurde 5.6V asemel näiteks SMAZ5V6-13-F, siis see on palju informatiivsem ja soovi korral võib alati igaüks selle spetsifikatsioonist ta parameetrid täpselt järgi vaadata.

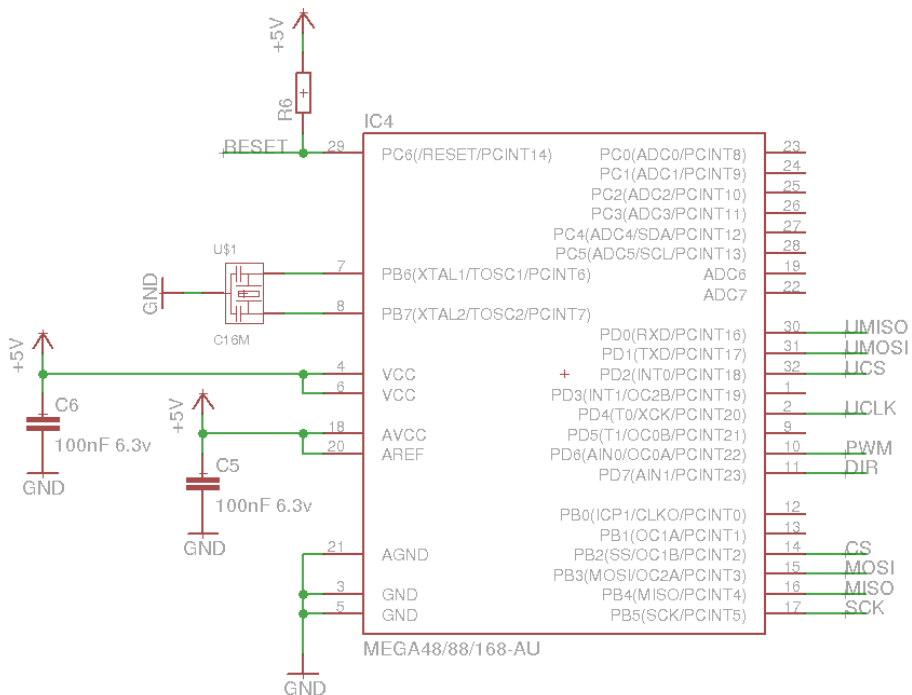


Ära joonista kõiki lahtisidestuskondensaatoreid skeemi ühte nurka kokku. Siis on nii endal kui teistel raske hoomata, et mille juurde nad täpsemalt kuuluvad.

GND

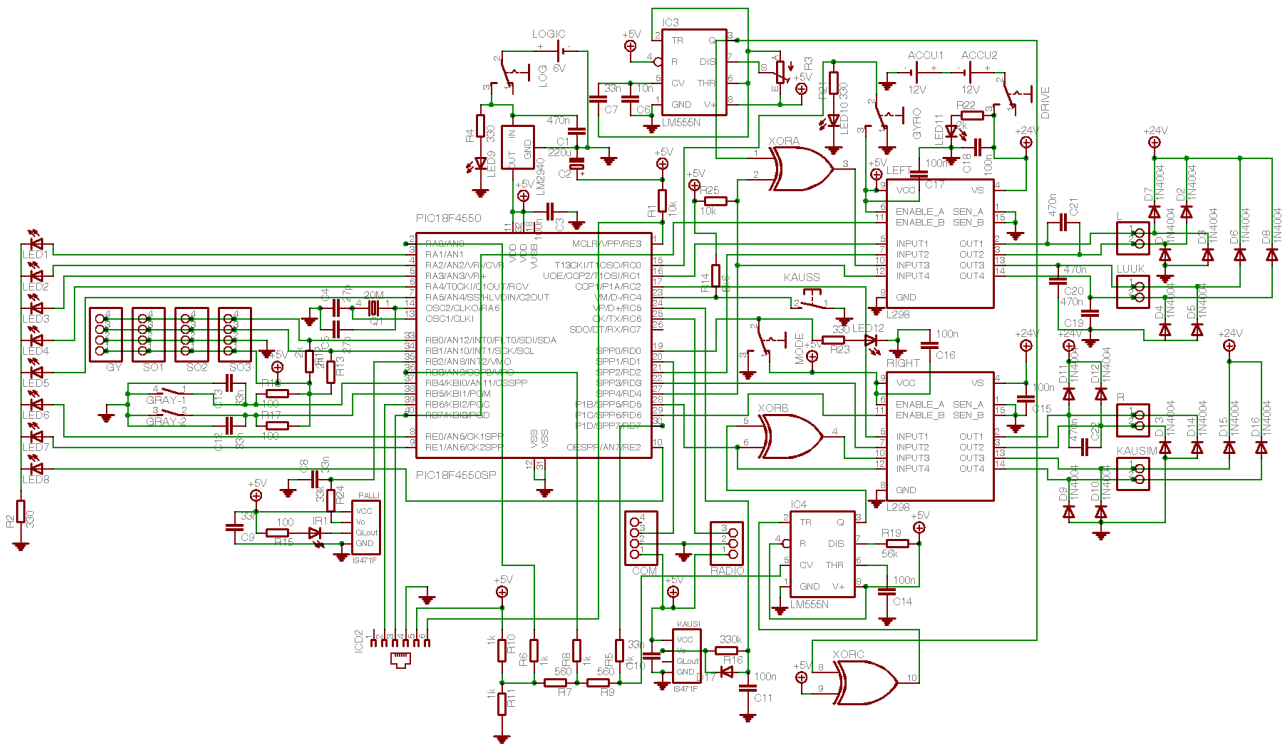


+VALE

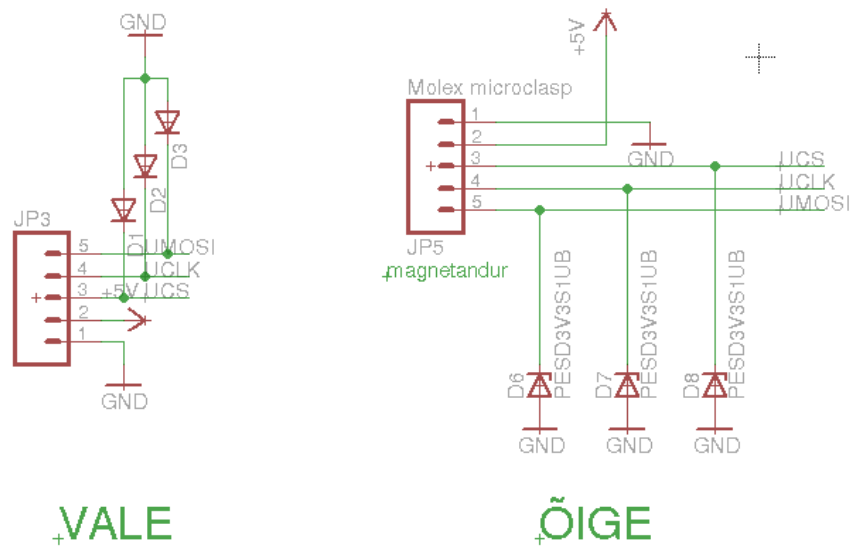


+ÕIGE

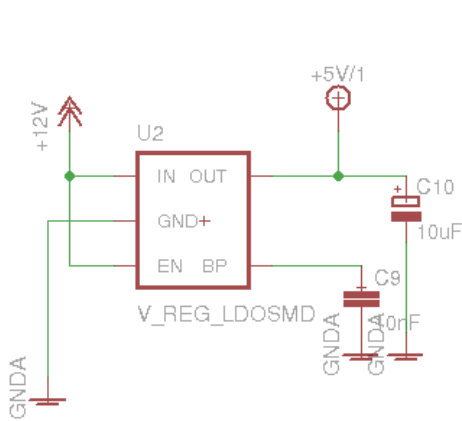
Ära ürita kõike võimalikult pisikeseks kokku suruda. Ära vea signaale risti üle komponentide.



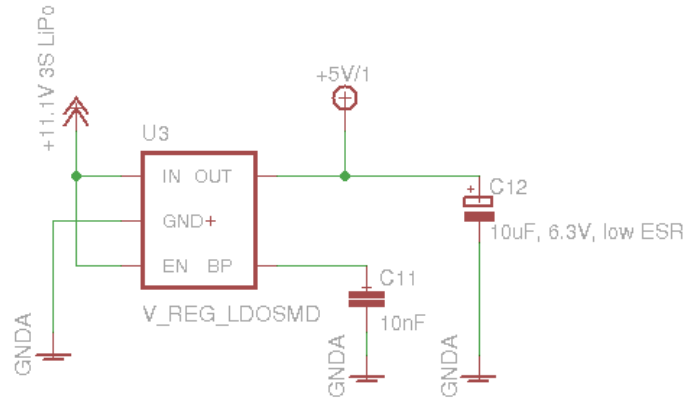
Pistikutele kirjuta juurde, kuhu nad lähevad või milleks nad on. Lisaks oleks hea ära märkida ka pistiku tüüp, et oleks lihtsam skeemi ja plaati võrreldes aru saada, et mis on mis. Lisaks lihtsustab see ka komponentide tellimisinimekirja koostamist. Võimaluse korral katsi maa joonistada suunaga allapoole ja toide ülespoole. Kasuta skeemil õigeid komponentide sümboleid (ära joonista stabilitroni asemel diodi jne).



Vajaduse korral lisa täpsustavat teksti seal, kus midagi on vaja ekstra silmas pidada. Kui toiteks kasutatakse näiteks 3-elementilist LiPo akut, siis kirjuta toitepingeks +12V asemel antud aku nominaalpinge +11.1V ja märgi juurde, et see tuleb 3S LiPo pealt. Kui see skeemil liiga palju ruumi võtab, siis märgi vähemalt toitepistikute juurde ära, et toide võetakse 3S LiPo't, mitte ei ole stabiliseeritud +11.1V

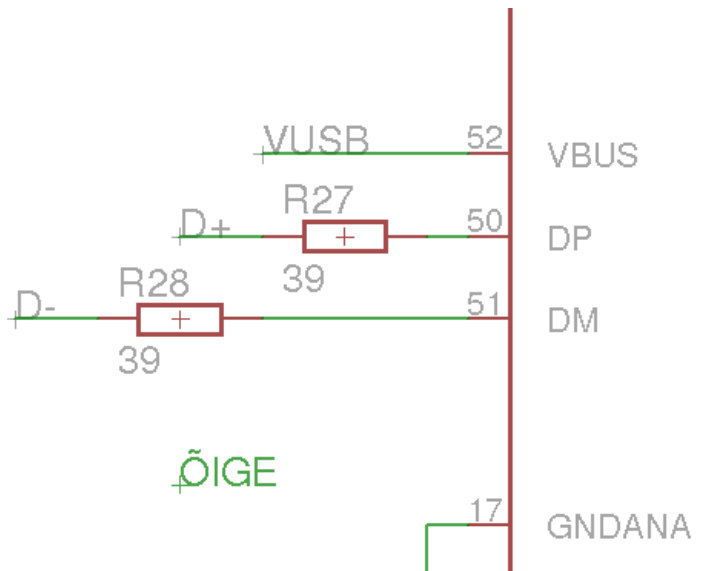
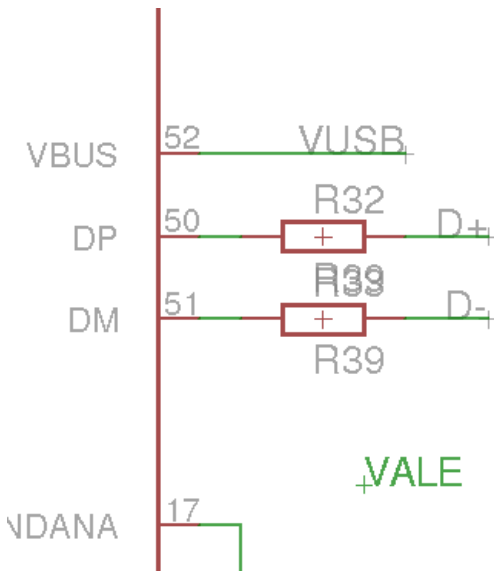


+VALE

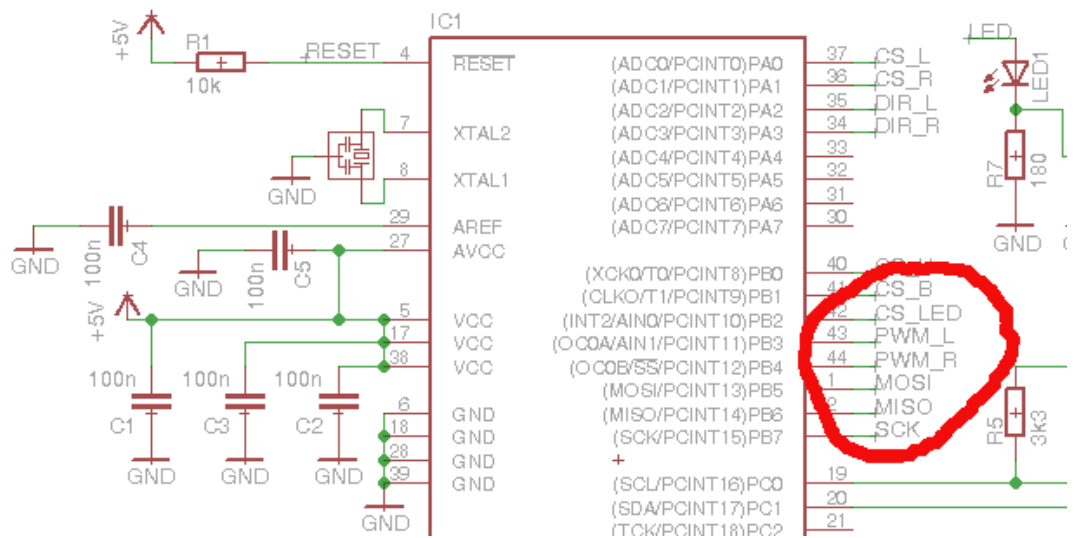
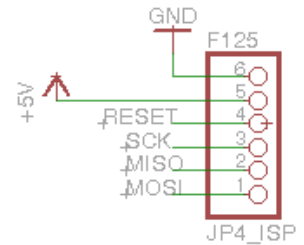
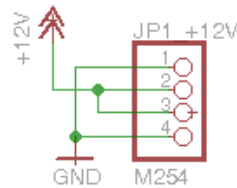
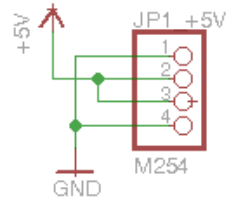
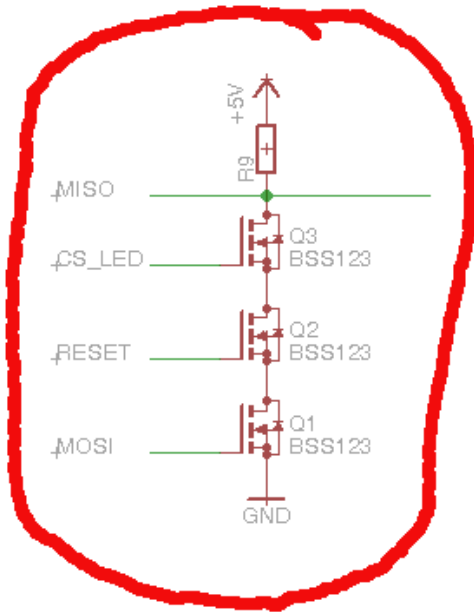


+ÕIGE

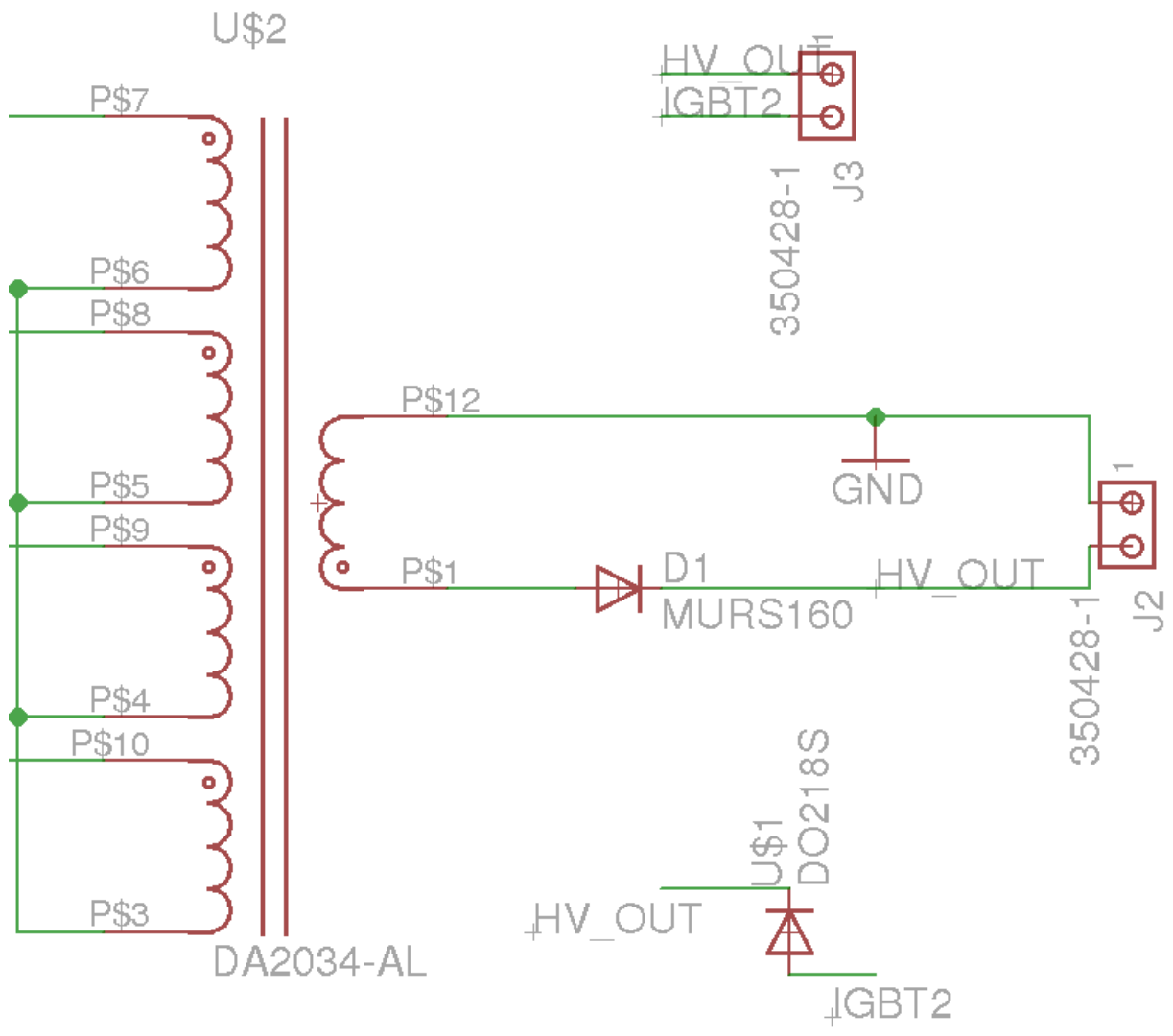
Vaata ka ise loogiliselt, et skeemist oleks võimalik aru saada. Kui ühe takisti juurde kirjutada R32 ja R39, siis see tekitab segadust. Takistitel kokkuleppeliselt oomi-märki taha ei joonistata, vaid ilma ühegi täheta number tähendabki oome ilma igasuguse kordajata. Kondensaatorite puhul tähistatakse ilma ühikuta numbriga pikofaradeid, aga tihtilugu ei peeta sellest reeglist kinni. Kindluse mõttes on alati parem kirja panna rohkem kui vähem.



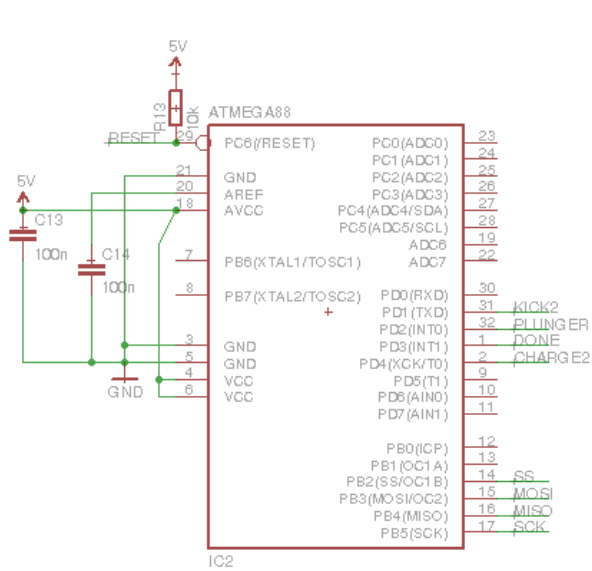
Katsu skeemil asju loogiliselt paigutada, et omavahel seotud asjad ka paikneks lähestikku. Katsu järgida head tava, et sisendid on vasakul, väljundid paremal. Samuti toide üleval ja maa all.



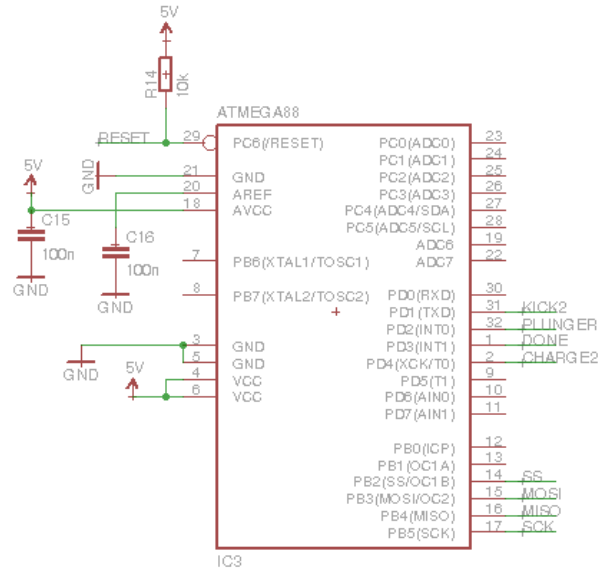
Digitaalelektronikas on vahel vaja viia paralleelselt palju signaale sama teed mööda. Näiteks aadressi- või andmesiin. Selleks kasuta Eagle „bus“ käsku. Kuidas täpsemalt, vaata näiteks youtubest: <http://www.youtube.com/watch?v=3c4PcVPb2hQ> Igat signaali ei pea alati otsast otsani joonega välja joonistama. Skeem muutub palju loetavamaks, kui ära näidata ainult signaali algus ja lõpp, mõlemale anda sama nimi ning see nimi ka skeemile kirjutada (vt. eelmisel pildil näiteks MISO, MOSI, RESET ja CS_LED signaale). Siiski ei maksa seda võimalust üle ekspluateerida. Alati tuleb kaaluda, kuidas skeem loetavam on – kas tervet signaali välja joonistades või signaali nimesid kasutades. Järgneval skeemil pole mingit mõtet HV_OUT ja IGBT2 signaale jätta täies ulatuses välja joonistamata. Selle asemel võiks pigem pistiku J3 ja diodi kõrvuti panna ja otse ühendada.



Skeemil ära kasuta mõjuva põhjusteta muid jooni peale horisontaalsete ja vertikaalsete. Ära püüa iga hinna eest kõiki maa- ja toiteühendusi kokku vedada.

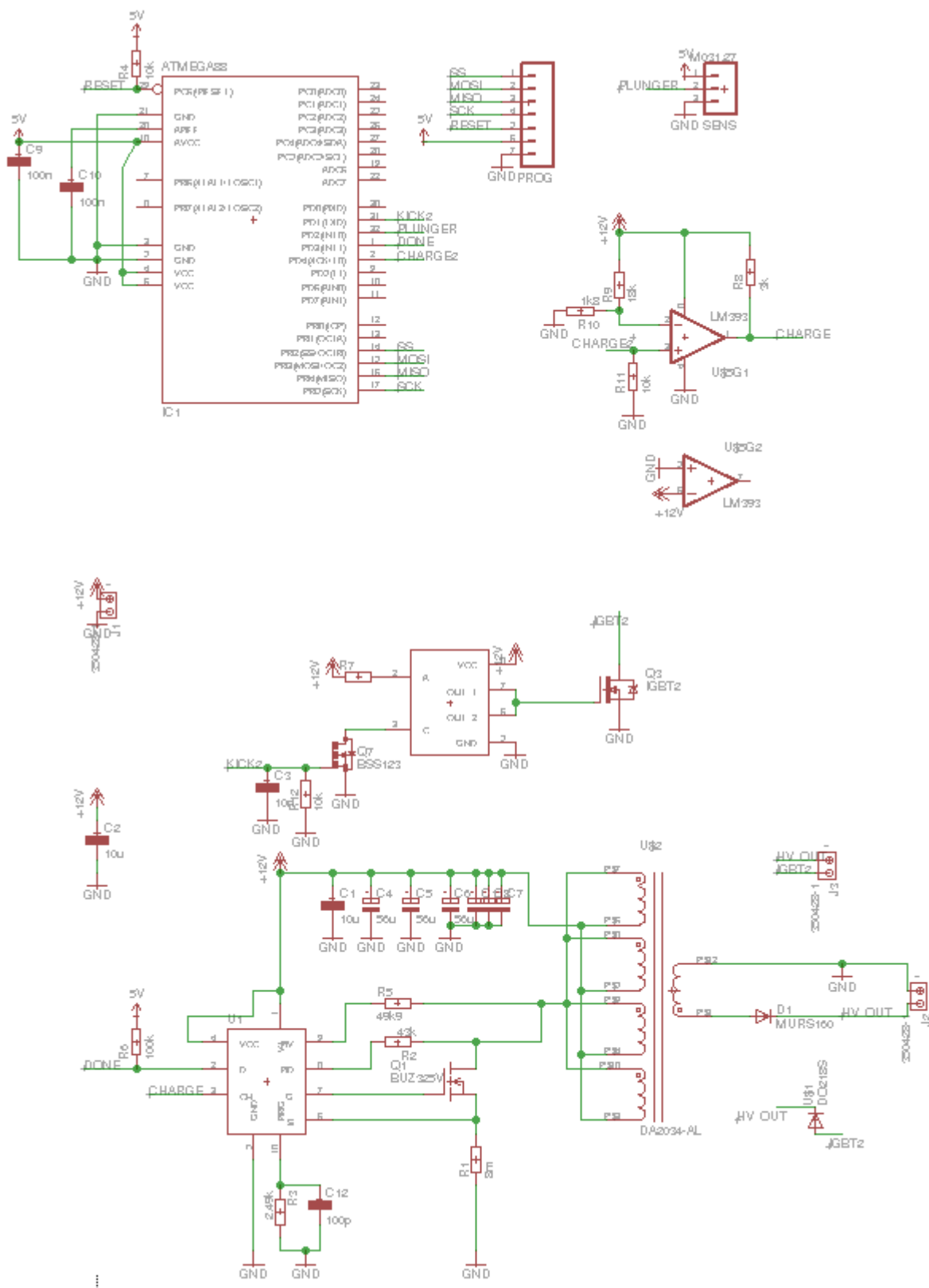


VALE

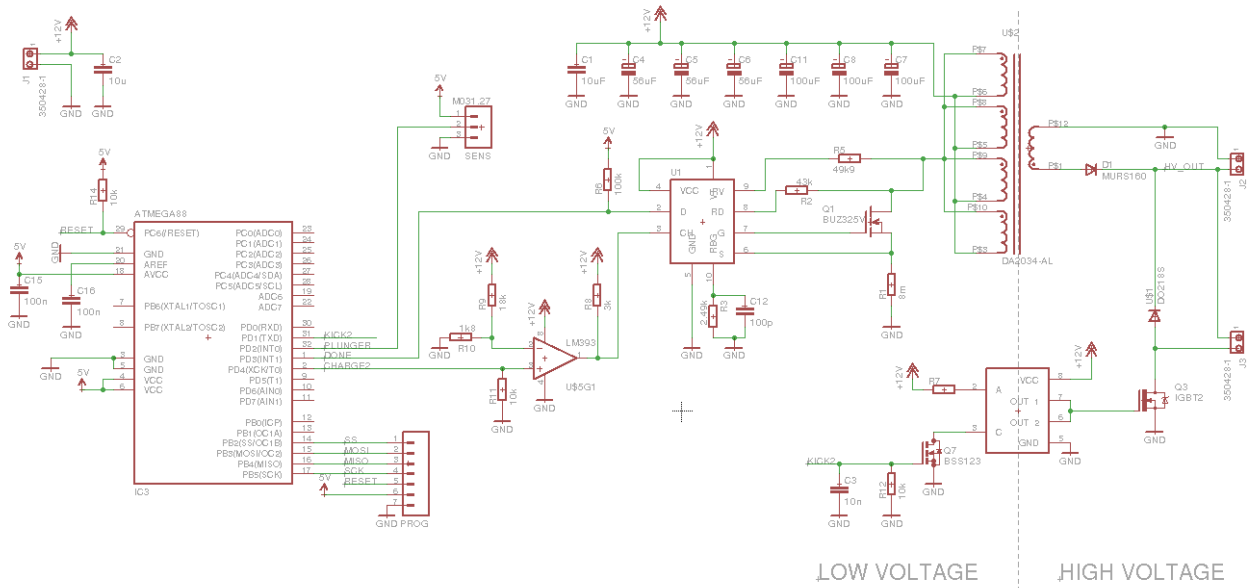


ÕIGE

Vaata seda digi wikist võetud *coilguni* skeemi ja mõtle, kas seda on mugav lugeda?



Nüüd vaata täpselt sama skeemi, ainult teistmoodi joonistatuna. Kumba varianti sa ise parema meelega vaataksid.





Sponsorlusest



Kellelt küsida?



Kellelt küsida?

- Esimesena tulevad pähe suured ja tuntud ettevõtted – EMT, Elisa, Saku Õlletehas, ...
 - Kahjuks tulevad need ettevõtted esimesena pähe ka kõigile teistele, kes sponsoreid otsivad – alustada võiks sponsorite otsimist vähem tuntud ettevõtetest



Kellelt küsida?

- Sponsoreid tasuks otsida ettevõtete hulgast, mille tegevus seostub valdkonnaga, mille jaoks raha küsitakse
 - Kui ettevõtted suudavad näha seost endaga (näiteks tulevaste töötajate, klientide näol), on nad paremini meelestatud sponsorluse osas
 - Kujutavad ette, mille jaoks raha kasutatakse (ülevalde)



Mida küsida?



Mida küsida?

- Kõik küsivad raha
 - Raha on üldiselt viimane asi, mida ettevõtte anda tahab
 - Raha asemel pakutakse enamasti allahindlust, kinkekaarte, kaupa – arvestage sellega (näiteks Brandnerist küsige trükkplaate, Oompoest elektroonikakomponente, ...)



Kuidas küsida?



Kuidas küsida?

- Välimus
 - Esmamulje on tähtis ja välimus on esimene asi, mida märgatakse. Puhtus on oluline.
 - Viisakus
 - Jälgige oma keelekasutust ja sõnavara – ropendamine ei tule kasuks
- Üldiselt – kui te sponsorile ei meeldi, siis on väga raske temalt midagi saada, olenemata sellest kui hea on robot



Kuidas küsida?

- Valmistuge sponsoriga kohtumiseks
 - Aega antakse teile vähe – kasutage seda hästi, tekitage sponsoris huvi ja nii võidate endale aega juurde
 - Valmistage end ette - slaidid, kõne, näidised
 - Ärge unustage end tutvustada ja seletada, mille jaoks sponsorlust küsitakse



Kuidas küsida?

- Näidised on head
 - Ainult jutt on igav – näidake asju (slaidid, pildid varem tehtud töödest, videod, töötavad robotid...)
 - Mittetöötavad lahendused jätke koju – ainult teie silmis on poolikud asjad midagi väärt. Sponsori jaoks on see tühi koht



Kuidas küsida?

- Jälgige oma esitlust
 - Tehnilised terminid ja lühendid unustage ära – kui sponsor teist aru ei saa, siis ta ei viitsi teid kuulata
 - Andke edasi asju ülevaاتlikult – graafikud, diagrammid on ülevaاتlikumad, kui tabelid, kirjutatud tekst.
 - Näidake pigem 3D mudelit kui tehnilisi jooniseid või programmi koodi 😊



Kuidas küsida?

- Olge konkreetsed

- Küsige konkreetseid asju – on suurem tõenäosus saada seda, mida küsite.

Näiteks:

- “Palju raha roboti ehitamiseks” vs
 - “3 Maxoni mootorit” või “1 ruutmeeter 3 mm paksust alumiiniumi”



Kuidas küsida?

- Olge valmis küsimustele vastama ja oma valikuid mõistlikult põhjendama
 - Ei tea või sellepärast, et oleks äge, on halvad vastused
- Ärge suhtuge küsimustesse üleolevalt või tooniga “iga lollakas peaks ju teadma”



Mida pakkuda?



Mida pakkuda

- Sponsorid ei anna asju tasuta:
 - Pakkuge vastu reklaami
 - Tavaline on logod särkidel, robotil, ...
 - Sponsori info robotiga seotud lehtedele (meeskonna leht, robotiklubi leht,...)
 - Arvestage sellega, et ettevõtte suudab ka ise reklaami osta – leidke ettevõtte jaoks ahvatlev pakkumine:
 - Näiteks youtube-i video, kus robotil on peal ettevõtte kaamera ja teie hea hinnang sellele kaamerale koos infoga sponsorile..



Mida pakkuda?

- Olge pakkumistes konkreetsetes:
 - “paneme ettevõtte logo oma särgile” – kas seda särki kantakse 1 kord Robotexil või iga päev niikaua, kui auklikuks kulub?
 - “lisame ettevõtte logo veebilehele” – kas see on seal siis igavesest ajast igavesti või kuni Robotexi lõpuni?



Mida pakkuda?

- Mõtelge mittestandardsetele lahendustele:
 - Reklaami pakuvad kõik, pakkuge midagi mida mujalt väga ei saa – näiteks ettevõttega kaasa messile nende boksi ringi sõitma robot ja kaasa inimene, kes roboti eest vastutab.



Mida pakkuda?

- Valmistage end ette küsimusteks:
 - Tüüpilised küsimused sponsorilt:
 - “Mitu inimest minu firma logo Robotexi lehel näeb?”
 - “Palju külastajaid on Robotiklubi veebilehel?”
 - Vastus: “ei tea” on halb
 - “Kui palju külastajaid on Robotexil, kes minu firma logo näeb?” - kui ei tea, siis öelge näiteks, et toimub ülekanne internetis ja kõik üle Eesti saavad seda näha, mitte ainult külastajad kohapeal



Mida veel?



Mida veel?

- Tundke sponsorit:
 - Ei maksa rääkida näiteks Saku Õlletehases, et Saku Originaal ei kõlba juua
 - Näidake ettevõtte vastu üles huvi



Mida veel?

- Isiklik lähenemine
 - Jätke sponsorile mulje – et olete spetsiaalselt tema välja valinud, mitte järjest läbi astunud kõigist tänaval ette sattunud firmade kontoritest
 - Parem on helistada ja ise kohale minna, kui e-maile kirjutada (neid ei viitsita lugeda)
 - Hoidke sponsoriga häid suhteid – nii on teinekord lihtsam sponsorlust saada



Mida veel?

- Õppige kaotama
 - Arvestage sellega, et iga küsimine ei anna tulemust, sellest hoolimata tuleb edasi proovida, ükskord ikka õnnestub
 - Analüüsige oma esitlust ja tehke vastavad korrektuurid



Mida veel?

- Õppige kaotama
 - Kui sponsor ei anna raha, siis tänage viisakalt leitud aja eest – kindlasti ei tohi hakata sõimama vms, sest
 - Ettevõtete juhid on üks seltskond ja ilmselt levib info halva käitumise kohta teistesse ettevõtetesse ja seal ei võeta teid pärast enam jutule



Meenutage oma ettekannet –
mida teeksite teisiti?

Tulemuste analüüsimine

Tähtis osa projekti juures on analüüsida projekti pärast selle lõppu – nii leiame üles mõistlikud lahendused, mida kasutada ka järgmiste projektide juures ja näeme probleeme, mis võivad tekkida ka järgmistes projektides ning olla nende lahendamiseks või vältimiseks valmis.

Tulemuste analüüsimise jagame etappidesse:

- Meeskonnatöö analüüs
- Lahenduse analüüs
- Projekti töö analüüs

Meeskonna analüüs

Meeskonna analüüsis meid huvitab, millised korralduslikud otsused (meeskonnatöö vahendite valik jms) aitab meeskonnatööl kaasa ning millised korralduslikud otsused osutusid ebamõistlikuks (st mis ei aidanud meeskonnatööl kaasa).

Uurime, millised suuremad probleemid meeskonnatöös esile kerkisid ning iga probleemi juures vaatame. Kas probleem õnnestus lahendada? Kui jah, siis kuidas probleem lahendati, kui ei, siis leiame põhjused, miks ei õnnestunud probleemi lahendada. Leida tuleks võimalused, kuidas selliseid probleeme tulevikus vältida.

Lahenduse analüüs

Lahenduse analüüsis huvitab meid, kas see lahendus, mille me valime ja otsustasime realiseerida tegelikkuses ka töötab. Analüüsis leiame vastused järgmistele küsimustele:

1. Kas lahendus õnnestus realiseerida täielikult (kogu ulatuses nagu planeeritud)? Kui ei, siis mis jäi realiseerimata ja miks?
2. Kas lahendus töötas nii nagu vaja? Kui ei, siis miks.
3. Kas lahendus rahuldab kõiki nõudeid nii nagu vaja? Kui ei, siis mis jäi rahuldamata ja miks?
4. Mõelda tuleks ka kliendi seisukohalt vaadates – kui projekti ellu viinud meeskond oleks projekti tellija, siis kas tulemusega ollakse rahul? Kui ei, siis mis jääb puudu?
5. Mis lahenduse juures töötas oodatust paremini? Mis lahenduse juures töötas oodatust halvemini?
6. Milliseid lahendusi kasutaksite järgmistes projektides, milliseid mitte ja miks?

Projekti töö analüüs

Projekti töö analüüsis huvitab meid eelkõige:

1. Kas ja kui palju oli plaanist kasu? Põhjendada
2. Kas meeskond suutis plaanist kinni pidada? Kui ei, siis miks?
3. Milliseid töid ei suutnud te ette näha ja plaani sisse arvestada? Kuidas seda teinekord vältida?

Vaatame ka ajahinnanguid. Eelkõige huvitab meid, milliste tööde puhul alahinnati tööle kuluvat aega? Kui palju oli vahe hinnatud ja tegeliku ajakulu vahel ja miks?

Analüüsi kokkuvõte

Analüüsi peab kirja panema ja vormistama korraliku dokumendina – see on üks alustuspunkt, millest järgmise projekti planeerimisel alustada.

1. Ülesanne – meeskonna wiki

Igal meeskonnal on oma töö korraldamiseks Moodle-s wiki. See wiki hakkab olema töövahend, kus meeskond hoiab oma projekti andmeid ja ülevaadet projekti hetkeseisust ning muud meeskonnatööga seotud informatsiooni.

Tekitada meeskonna wikisse lehekülg, kus on meeskonna andmed:

- Meeskonna nimi
- Meeskonna liikmete nimed
- Meeskonna liikmete kontaktandmed (e-mail, telefon)

2. Ülesanne – meeskonna analüüs

Analüüsida oma meeskonda.

1. Kirja panna iga meeskonnaliikme kohta vähemalt 5 oskust. Oskused tuleb kirja panna piisavalt detailselt (näiteks oskus joota SMD komponente), mitte liiga üldiselt (elektroonikaoskus).
2. Välja selgitada meeskonna tugevused ja nõrkused. Tulemus vormistada tabelina
3. Analüüsida meeskonna tugevuste ja nõrkuste tabelit. Kirja panna tabelist nähtavad probleemid ja ettepanekud nende probleemide lahendamiseks.

Lahendus vormistada korrektse dokumendina (rtf või pdf formaat) ning meeskonna analüüs lisada ka wikisse.

4. Ülesanne – projekti etapid

1. Koostada projekti elutsüklil oma robootikaprojekti kohta. Iga etapi kirjelduses lähtuda oma projekti ülesande püstitusest.
2. Iga etapi juures tuua välja küsimused, millele ei tea vastust, aga mis tuleb projekti edukaks lõpetamiseks välja selgitada.

Tulemus vormistada korrektse dokumendina (rtf või pdf) ja elutsüklil kirja panna ka oma meeskonna wikisse.

5. Ülesanne – projekti õnnestumine

1. Panna kirja kuni 5 suuremat põhjust, miks käesoleva kursuse raames meeskonna poolt tehtav projekt võib ebaõnnestuda.
2. Iga põhjuse juurde lisada meetmed, mida rakendate, et ebaõnnestumisi vältida.

Lahendus vormistada korrektse dokumendina (rtf või pdf).

6. Ülesanne – nõuded

Analüüsida oma projekti ja panna kirja projekti nõuded. Nõuded jagada omal valikul gruppidesse (näiteks nõuded mehaanikale, nõuded elektroonikale, ... või nõuded liikumismoodulile, nõuded ümbruse tajumisele, ... vms).

Iga nõude juurde märkida ka klassifikatsioon – kas nõue on:

- kohustuslik – kui nõue pole täidetud, siis robot ei täida etteantud ülesannet
- soovituslik – kui nõue pole täidetud, siis robot täidab etteantud ülesannet, kuid mitte nii kiiresti/osavalt kui võiks
- mugavus – kui nõue pole täidetud, siis robot täidab etteantud ülesannet, kuid robotit pole kasutada nii mugav

Nõuded vormistada tabelina, iga nõude juurde kirjutada, kuidas kavatsete nõude täidetust kontrollida.

Lahendus esitada korrektse dokumendina (rtf või pdf) ning üles panna ka meeskonna wikisse.

7. Ülesanne – kliendiga suhtlemine (olukord 1)

Firma X saadab oma kontorist A teises linna otsas asuvasse kontoris B taksoga paki. Kontorist B teatakse, millal pakk saabub, aga ometi peab taksojuht väljas 10 minutit ootama. Kontori B juures on tihti probleem, et pole lihtne leida parkimiskohta.

Taksojuht viskab paki kontorist B taksost pakile vastu läinud töötajale ja lisab ärritunult: „Ma olen siin juba tükk aega oodanud! Selle firmaga siin on kogu aeg probleeme.“

Selle peale ärritub ka firma X töötaja ja helistab taksofirmasse, et teha kaebus ja järgmine kord ta enam taksot ei telli.

Eeldage, et teie olete taksojuht ja teie kliendiks on firma X.

1. Mida tehti valesti?
2. Kuidas seda olukorda mõistlikult lahendada?

8. Ülesanne – kliendiga suhtlemine (olukord 2)

Firma X tellib firmalt Y roboteid. Firma Y töötab roboteid välja ja valmistab neid. Firma Y on lubanud firmale X viie päeva pärast anda kolm uut robotit, mis veel pole valmis, aga mille kallal tehakse kõvasti tööd ja kõik töötajad on robotitega hõivatud, et õigeaegselt valmistada

jõuda. Firma Y ei ole huvitatud lisatööde tegemisest, kartes et robotid ei saa siis õigeaks ajaks valmis.

Firmast X helistatakse firmasse Y: „Me ei oska robotit tööle panna ja meil on kohe vaja, et te tulete meile siia seda robotit tööle panema. Kogu meie firma töö selle roboti tõttu praegu seisab.“

Firmast Y uuritakse: „Kuidas see juhtus, et te robotit tööle panna ei oska – siiani ju olete sellega alati hakkama saanud?“

Firma X: „Jah, aga varem oli meil siin üks inimene, kes roboteid tööle pani ja täna ta meil enam ei tööta.“ Selle töötaja lahkumine firmast X oli ette teada juba kuu aega.

1. Eeldage, et olete firma X töötaja. Kuidas lahendaksite olukorra? Kuidas kindlasti käituda ei tohiks?
2. Eeldage, et olete firma Y töötaja. Kuidas lahendaksite olukorra? Kuidas kindlasti käituda ei tohiks?

9. Ülesanne – kliendiga suhtlemine (olukord 3)

Firma U tahab tellida firmalt K tarkvara, mis leiab inimestest tehtud fotodelt üles näo ja ütleb selle järgi inimese vanuse. Firma U väidab, et selle tööga on kohutavalt kiire ja tahab teada, kas seda on võimalik teha.

Firma K väidab, et sellist tarkvara saab teha ainult siis, kui pildil on üks inimene ja temast on tehtud pilt eest otsevaates.

Firma U ütleb, et seda tarkvara on kohe arendama vaja hakata ja see peab 3 nädala pärast valmis olema. Firma K küsib algandmeid – fotosid inimestest. Firma U pärast poolenädalast venitamist saadab mõned pildid, mis on tehtud inimeste jalgadest. Firma K selgitab, et need pildid on valed ja küsib uusi. Siis firma U vastab – aga hankige need õiged pildid siis ise. Aga tarkvara peab siiski õigeaks ajaks valmis saama (kuigi algandmete otsimisega kulutati nädal).

1. Eeldage, et olete firma K töötaja. Kuidas lahendaksite olukorra? Kuidas kindlasti käituda ei tohiks?

10. Ülesanne – meeskonnatöö korraldamine

Panna paika meeskonnatöö korraldus:

1. Jagada ära meeskonnaliikmete vahel rollid ja vastutusvaldkonnad.
2. Valida välja vahendid, mida kasutatakse meeskonnatöös. Kui valitud vahendid eeldavad seadistamist, siis vajalikud keskkonnad üles seada.
3. Panna paika meeskonnatöö reeglid.

Lahendus esitada moodle-s ja kajastada ka meeskonna wikis.

11. Ülesanne - tasklisti täitmine

Vastavalt koostatud projektiplaanile jagada ära meeskonna sees ülesanded ja need ülesanded kajastada tasklistis vastavalt juhendile.

12. Ülesanne - lahenduse üldkirjeldus

Koostada lahenduse üldkirjeldus. Aluseks tuleb võtta varem koostatud nõuete dokument ja lisada sellele veerud:

- võimalikud lahendusvariandid (1-2 tk)
- vajalikud ressursid (materjalid)
- vajalikud ressursid (oskused/teadmised)

Tulemus vormistada korrektse dokumendina (rtf või pdf) ja esitada keskkonnas Moodle.

13. Ülesanne - mehaanika lahenduse kirjeldus

Koostada mehaanika lahenduse kirjeldus oma projekti jaoks. Kirjelduses peab sisalduma:

- mehaanika üldkirjeldus
- moodulite loetelu
- mooduleid ja nende vahelisi seoseid iseloomustab koostediagramm.

Lahenduses vajalike diagrammide koostamiseks võib kasutada näiteks programmi MS Visio. Uurida lisaks SysML-i (see on UML-il baseeruv vahend tehniliste lahenduste dokumenteerimiseks).

Lahendus vormistada korrektse dokumendina (rtf või pdf).

14. Ülesanne - elektroonika lahenduse kirjeldus

Koostada elektroonika lahenduse kirjeldus oma projekti jaoks. Kirjelduses peab sisalduma:

- elektroonika üldkirjeldus
- moodulite loetelu
- mooduleid ja nende vahelisi seoseid iseloomustab seoste diagramm.

Lahenduses vajalike diagrammide koostamiseks võib kasutada näiteks programmi MS Visio. Uurida lisaks SysML-i (see on UML-il baseeruv vahend tehniliste lahenduste dokumenteerimiseks).

Lahendus vormistada korrektse dokumendina (rtf või pdf).

15. Ülesanne – tarkvara lahenduse kirjeldus

Koostada tarkvara lahenduse kirjeldus. Selles peab sisalduma:

- tarkvara üldkirjeldus
- käitumisloogika diagramm
- tarkvara jagamine kõrgema ja madalama taseme loogikaks, kusjuures kummagi grupi jaoks kirjeldada ära selles kasutatavad funktsioonid.

Lahenduses vajalike diagrammide koostamiseks võib kasutada näiteks programmi MS Visio.

Lahendus vormistada korrektse dokumendina (rtf või pdf).

16. Ülesanne – lahenduse analüüsimine

Analüüsida oma projekti. Analüüsis tuua välja:

1. lahenduse vastavus nõuetele
2. lahenduse teostatavus
3. lahenduse keerukus
4. lahenduse modulaarsus
5. lahenduse testimine
6. lahenduse kulukus

Leida oma projektis üles riskid ja pakkuda välja võimalused riskide maandamiseks. Riskid jaotada:

1. Riskid meeskonnast
2. Riskid lahenduse valikust
3. Riskid tarnijatest
4. Muud riskid

Lahendus vormistada korrektse dokumendina (rtf või pdf).

17. Ülesanne – eelarve

Koostada roboti jaoks vajalike detailide nimekiri ja hinnang eelarvele.

Nimekirjas kindalsti eraldada välja elektroonika ja mehaanika detailid. Iga detaili kohta kirjutada täpsustused:

- kogus
- hind
- kust tellida/osta või valmistada
- kaua tellimine/valmistamine aega võtab
- valmistatavate detailide korral materjalid ja hinnang materjalikulule
- uurida kust materjali saab ja mis hinnaga
- jne..

Otsida välja ja kirja panna ka mõistlikud alternatiivid (näiteks mootorite korral jms korral).

Tuua eraldi välja juba tellitud detailid, tarnijad, tarneajad ja -tähtajad, hinnad ja kogused.

Dokument korrektselt vormistada (lisada dokumendile ka tiimi nimi ja/või liikmete nimed)

18. Ülesanne – ülevaatlik projektiplaan

Koostada ülevaatlik projektiplaan. Plaanis peab olema näha:

1. Suuremad tööde grupid ja nende gruppide vahelised sõltuvused (ka ajalised sõltuvused)
2. Vaheetapid ja iga vaheetapi oodatavad tulemused
3. Tööde gruppide prioriteetsus

Projektiplaan tuleb vormistada korrektse tabelina ja üles panna oma meeskonna wikisse. Seda plaani tuleb jooksvalt edasise töö käigus muuta - tehtud tööde grupid ära märkida, samuti vajadusel muuta tuleviku plaane.

19. Ülesanne – elektroonikaskeemide disainimine (AVR32, draiverite plaat)

AVR32 plaat ja draiverite plaat

1. Tutvuta olemasolevate elektroonikaskeemiga
2. Leppida mehaanikutega kokku plaatide dimensioonid
3. Disainida AVR32 plaat
4. Disainida draiverite plaat
5. Analüüsida skeemi, otsida vigu ja parandada
6. Dokumenteerida skeemid

Lahenduseks esitada eagle-i failid (skeem ja plaadi disain) ning dokumentatsioon.

20. Ülesanne – elektroonikaskeemide disainimine (mootorite plaat)

Mootorite plaat

1. Tutvuda olemasoleva skeemiga ja parandusettepanekutega
2. Mõelda ja uurida ülejäänud tiimidel vajadusi mootorite plaadi täiendamiseks ning need dokumenteerida.
3. Parandused skeemi ja plaadi disaini sisse viia
4. Dokumenteerida skeem ja tehtud muudatused

Lahenduseks esitada eagle-i failid (skeem ja plaadi disain) ning dokumentatsioon. Tööd juhendab ja kontrollib Jaas.

Parandusettepanekud:

1. lisada plaadile pistikuid, mis tooks välja jalgu, mida andurite jaoks kasutada
2. magnetandurite spi transide asi parandada
3. vahetada välja prose mingi sellise vastu, mis on USB kaudu progetav (otsida välja mingi odavam variant)

21. Ülesanne – elektroonikaskeemide disainimine (Coil-gun)

Coil-guni elektroonika

1. Tutvuta olemasoleva elektroonikaskeemiga ja näpunäidetega
2. Vana plaati analüüsida (St kõik vead kirja panna (kellel mis ja millal läbi põles ja mis ei töötanud jms) ja mingit olemasolevat plaati mõõta).
3. Parandused skeemi ja plaadi disaini sisse viia
4. Dokumenteerida skeem ja tehtud muudatused

Lahenduseks esitada eagle-i failid (skeem ja plaadi disain) ning dokumentatsioon. Tööd juhendab ja kontrollib Erki.

Mõned näpunäited, mida võiks uurida ja võib-olla rakendada:

- 1) Laadimise ja juhtimise osa võiksid eraldatumad olla. Praegusel plaadil on kõik jupid ühel pool, parem võiks olla kui prose, IGBT driver jms oleksid teisel pool plaati või siis hoopis eemal.
- 2) IGBT driveril ei ole hetkel bypass kondekaid, mis peaksid olema.
- 3) IGBT'le gate takisti ja kas TVS või 2 zenerit kaitseks.
- 4) Kui laadimisvoolu piirata väiksemaks, siis on kindlasti ka müra väiksem.
- 5) Vtrans bypass kondekad peavad kindlasti olema low ESR.
- 6) Võib proovida Vtrans ja muu toite eraldamist kondeka ja pooliga (decoupling).
- 7) Rada IGBT gate driverist gateni peaks olema laiem ja lühem, kuna seal

võib maksimaalne vool päris suur olla

8) SPI pistik väga suurte vooludega raja kõrvale panna (nagu hetkel) on suht halb.

9) Pöörata suurt tähelepanu ruutimisel!

10) teha nii, et särtsu ei saaks - kõrgepinge võiks olla groundi suhtes, mitte 12V suhtes

Kõigepealt skeem:

1. Üldine: kõigi skeemide toitejalgade peal peaks olema keraamiline kondekas (100 nF on üldjuhul piisav). Muidu võib tekkida ebastabiilsusi.

2. IGBT draiveri juures võiks kasutada sellist skeemi, nagu datasheetis soovituslik (peaks vähendama igasuguseid probleeme).

3. ~~IGBT-ga paralleelselt peaks panema välise dioodi (selle sisemine ei pruugi olla piisav)~~. Lisaks pole paha mõte panna TVS diood IGBT gate ja maa vahele (kui küll boardi layout korda teha, siis on vajadus selle järele väiksem).

4. Atmega toitejalgadele keraamilised kondekad.

5. Signaalidel KICK2, DONE, PLUNGER ja CHARGE2 võiks olla takistid jadamisi sees (et vältida protsessori kahjustumist vea olukorras). lk peaks hea olema.

6. LM393 kasutamine level converterina on veidi overkill. Lihtsam oleks signaali tase inverteerida ja kasutada ühte FETi.

7. HV_OUTi ja maa vahel peaks olema üks keraamiline kondekas (kasvõi väike).

8. Soovitaks vaadata, kas elektrolüüdid saab asendada tantaalidega (nii võib õnnestuda skeemi pidndala vähendada ja töökindlust tõsta). Tõenäoliselt on ka võimalik täiesti keraamiline lahendus, mis võiks olla müra seisukohast parem, kuid see võib tarbetult kalliks minna.

9. +5V ahelas võiks olla rohkem mahtuvust. (Üks >10 uF kondekas võiks ka olla.)

PCB:

Kõige suuremad probleemid on PCB layoutiga. Muuga pole nii kriitiline, kuid suure võimsuse ja suure sagedusega ahel tuleks kompaktsemalt paigutada ja eraldada andmetöötluslikust osast.

Ma arvan, et kõige lihtsam oleks PCB disaini algusest peale uuesti teha.

1. Soovitaks jälgida trafo draiveri juures selle soovituslikku layouti. Igastahes peaks kogu sellega seotud skeem olema trafote võimalikult lähedal. Hetkel on Atmel ja IGBT draiver keset seda, mis on enam-vähem kõige halvem koht nende jaoks antud plaadil üldse. Trafo draiver ja kondekad peaks ikka olema nii lähedal trafote kui võimalik. Soovitaks paigutada kondekad otse trafo jalgade peale. Muidu võiks jälgida draiveri

soovituslikku layouti datasheetist.

Kõik kondeka laadimisega otseselt mitte seotud signaalid peaks sellest osast ringiga ümber viima (nt IGBT draiveri signaal).

2. Ground plane: võimalikult suure osa skeemi mõlemast poolest vüiks täita ground plane (teha pollügonid, mis katavad mõlemad pooled). Tuleks ka ground plane-de vahele panna võimalikult palju gnd viasid (suurusjärgus ruutsentimeetri kohta üks või tihedamalt).

3. Thermal reliefid tasuks polügoni tekitamisel välja jätta. Nii ei teki neid suure takistusega kitsaid signaale trafo padide juurde.

4. Praktilise tootmise seisukohast soovitan teha nurgad 45 kraadi täiskordsetena. (Selleks on ka DRC-s check angle.) See on hea tava ja lisaks võib muidu teoreetilisel tootmisel probleeme tekkida.

22. Ülesanne - AVR-iga tutvumine

- a) Panna kokku makettplaadi peal skeem, mis koosneb AVR-st, LED-ist ja takistist.
- b) Koostada C-s programm, mis vilgutab LED-i sagedusega 1 Hz. Abimaterjaliks on datasheet (märksõnad: pordid, IO-space, DDRx, PORTx) ja AVR-libC API (märksõnad: `_delay_ms`).
- c) Koostada C-s programm, mis vilgutab morse koodis „S M S“-i korduvalt.
- d) Lisada makettplaadile nupp, mis on ühendatud IO sisendiga ning mille seisundit on võimalik AVR-iga tuvastada.
- e) Koostada C-s programm, mis vilgutab LED-i sagedusega 1 Hz juhul kui nupp on alla vajutatud. Kui nupp on üleval, siis LED ei põle.

Arvestuse saamiseks demoda töötavaid lahendusi punktidele a) – e).

23. Ülesanne - AVR.1

- a) Uurida katkestusi (interrupts: timer0 compare match interrupt, UART data register empty, UART receive complete). Arvestuse saamiseks vestlus Jorma ja Arturiga, kus peab tõestama, et teemast on aru saadud.
- b) Koostada programm, mis taimeri katkestusi kasutades saadab peatsüklist iga 0,5 sekundi tagant „Hello world!\n“. Arvestuse saamiseks demoda töötavat lahendust ja koodi. Vihjed: timer compare match interruptist pannakse püsti lipp, seda lippu kontrollitakse peatsüklist; volatile keyword C-s.
- c) Koostada ülesande b) osa baasil programm, mis kasutab UART katkestusi info saatmiseks. Arvestuse saamiseks demoda töötavat lahendust ja koodi. Vihjed: tekitada puhver, kuhu string

pannakse taimeri katkestuses; UART data register empty katkestuses võetakse puhvrilt baithaaval andmed ja saadetakse üle UARTi.

24. Ülesanne – AVR.2

Uurida FIFO (queue) andmestruktuuri kohta. Realiseerida FIFO, kuhu saab panna baite ja kust võetakse saatmiseks baite UARTi katkestuses. Täiendada ülesannet c) nii, et FIFO-sse lisatakse andmeid mitte ainult taimeri katkestuses, vaid ka peatsükli. Arvestuse saamiseks demoda töötavat lahendust ja koodi. Vihjed: CLI ja SEI instruksioonid.

25. Ülesanne – AVR32

a) Teha demoboardile programm, mis vilgutab LED-i. Arvestuse saamiseks näidata ette kood ja demoda töötavat lahendust.

b) Confida power manager nii, et protsessor töötaks sagedusega 30 MHz. Arvestuse saamiseks demoda töötavat lahendust - vilgutada LED-i sagedusega 1MHz.

c) Teha demoboardile programm, mis saadab üle UARTi peatsüklilist katkestusi kasutamata stringi: „foo bar – Jorma lemmik“. Arvestuse saamiseks demoda töötavat lahendust.

d) Uurida INTC (interrupt controller) tööd, confirmist ja kasutamist. Teha ülesanne c) ümber katkestusi kasutades. Uurida, kuidas on ARV32 software frameworkis tehtud katkestuste kasutamine ja välja tuua selle positiivsed ja negatiivsed küljed. Soovitada parem lahendus. Arvestuse saamiseks demoda töötavat lahendust ja koodi. Vihjed: linker skript, exception.S fail.

26. Ülesanne – jootmine

Joota kokku oma projekti jaoks vajalikud elektroonikaplaadid. Jootmiseks vajalikud vahendid leiab laborist.

27. Ülesanne – suhtlusprotokoll

Implementeerida suhtlusprotokoll:

http://digi.physic.ut.ee/mw/index.php/Sise:Protokoll_hnelix. Arvestuse saamiseks näidata ette kood ja demoda protokollitööd.

28. Ülesanne – roboti mehhaanika mudel

Üles tuleb laadida roboti koostejoonis, mis on korralikult vormistatud.

Joonis peab sisaldama roboti 3 vaadet ning läbilõikeid tähtsamatest sõlmedest. Tükitabelis peab olema täidetud materjalid (materjal, mõõdud, kogus).

29. Ülesanne - mootori juhtimine

Kasutades ülesande 7 ja 8 lahendust juhtida brushless mootori rootorit nii, et see kordab vaheldumisi samme 1. ja 2.:

1. hoiab rootori nurka konstantsena 1 sekundi jooksul
2. 0,5 sekundi jooksul teeb 72-kraadise pöörde.

Arvestuse saamiseks demoda töötavat lahendust.

30. Ülesanne - MLX90316SPI kasutamine

Panna tööle MLX90316SPI magnetpöördeandur. Selgitada välja selle anduri maksimaalne reaalne töökiirus ning kasutada seda. Tuvastada selle anduri järgi mootori rootori nurk staatori suhtes. Arvestuse saamiseks näidata töötav lahendus, mis näitab nurka.

31. Ülesanne - brushlessid

Teha selgeks brushless mootori juhtimise elektroonika. Panna kokku skeem makettplaadi peal ja juhtida selle abil kõvaketta mootorit suvalisel etteantud kiirusel. Kiirust peab saama programmist ette anda. Arvestuse saamiseks esitada kood, näidata ette skeem ja demoda töötavat lahendust.

32. Ülesanne - elektroonika debugimine

Debugida oma robotiprojekti elektroonikat, leida vead ja need ära parandada. Leitud vigade kohta teha analüüsi:

- millised vead leidsite
- kuidas vead avaldusid
- milliseid vigade leidmise meetodeid kasutasite
- kuidas vea parandasite

Kogutud analüüs panna üles oma meeskonna wikisse.

33. Ülesanne - lihtne sõitmine

Läbida robotiga trajektoor 1 -> 2 -> 3.

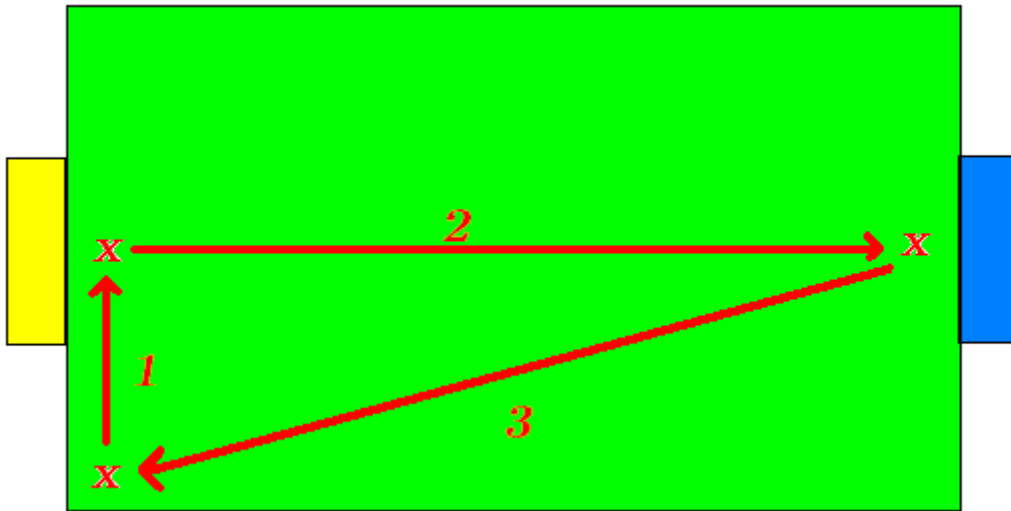
1 - nurgast sõita enda värava ette

2 - sõita enda värava eest vastase värava ette

3 - sõita vastase värava eest nurka, millest alustati.

Väravasse sõita ei tohi.

Stardipositsioon ja positsioonid väravate ees on märgitud väljakule valge paberteibiga.



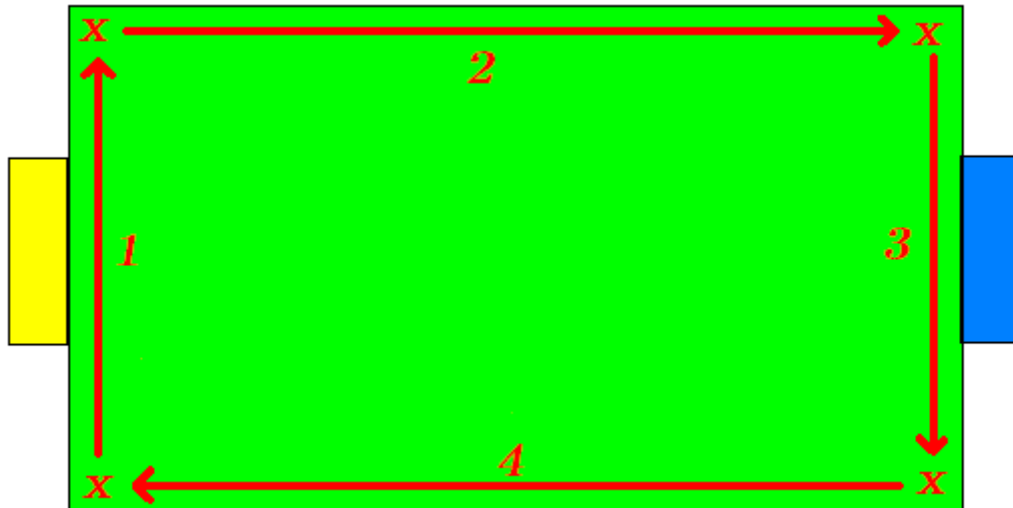
Hindamine:

- 1) Iga etapp 1, 2, 3 annab 1 punkti
- 2) Kogu trajektoori 1-2-3 läbimine annab ühe punkti
- 3) Ajalises järjestuses 2 kiireimat saavad lisapunkte vastavalt 1 ja 0,5 punkti.

Arvestuse saamiseks demoda töötavat lahendust. Demo jaoks on igal võistkonnal 2 võimalust (st kui esimese korraga ei õnnestu, siis võib uuesti proovida, arvesse läheb parim katse). Demo käigus robotit puudutada ei tohi (kui mõni võistkonnaliikmetest puudutab robotit demo käigus, siis loetakse demo lõpetatuks).

34. Ülesanne - mööda väljaku servasid sõitmine

Läbida väljakul trajektoori 1 -> 2 -> 3 -> 4. Roboti esiosa peab olema alati sõidu suunas. Sõitmist alustatakse nurgast. Väljaku servasid puudutada ei tohi ja väravasse sõita ei tohi.



Hindamine:

1) igakorrektselt läbitud trajektoori osa 1, 2, 3, 4 annab ühe punkti (seina puudutamisel või väravasse sõitmisel punkti ei saa).

2) 1 punkti saab lisaks kui läbiti korrektselt kogu trajektoori 1 -> 2 -> 3 -> 4.

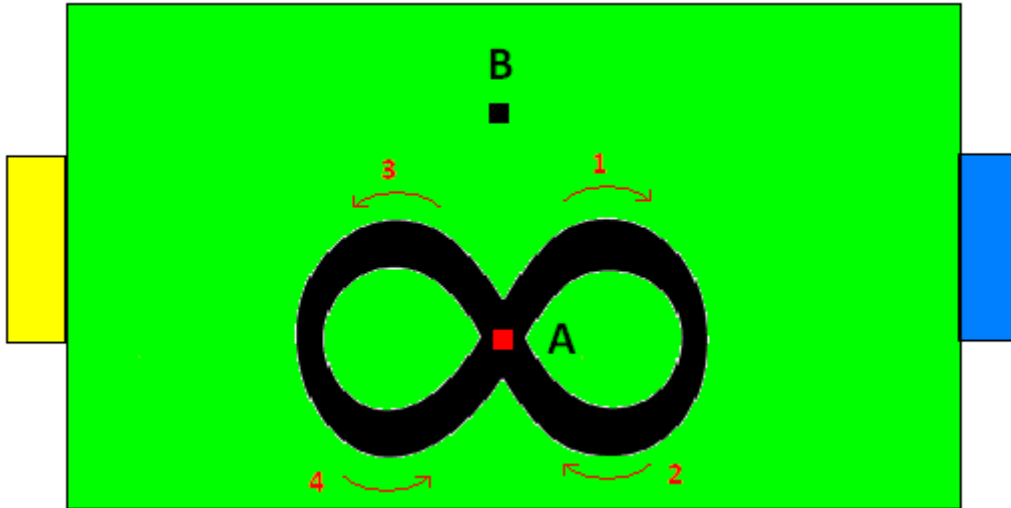
3) ajaliselt 2 kiiremat saavad lisapunkte vastavalt 1 ja 0,5 punkti.

Arvestuse saamiseks demoda töötavat lahendust. Demo jaoks on igal võistkonnal 2 võimalust (st kui esimese korraga ei õnnestu, siis võib uuesti proovida, arvesse läheb parim katse).

Demo käigus robotit puudutada ei tohi (kui mõni võistkonnaliikmetest puudutab robotit demo käigus, siis loetakse demo lõpetatuks).

35. Ülesanne – lõpmatuse sõitmine

Sõita väljakul lõpmatuse märki mööda, roboti löögimehhanismi osa suunatud punkti B poole. Sõitmist alustada punktist A. Trajektoori tuleb läbida kaks korda järjest. Need robotid, mis ei kasuta omniwheele, ei pea hoidma suunda punkti B poole.



Hindamine (omniwheelidega robotite puhul):

- 1) lõpmatuse sõitmine kaks korda - 3 punkti.
- 2) löögimehhanismiga õige suuna hoidmine - 3 punkti.

Hindamine (kahe rattaga robotite puhul):

- 1) lõpmatuse sõitmine kaks korda - 6 punkti.

Arvestuse saamiseks demoda töötavat lahendust. Demo jaoks on igal võistkonnal 2 võimalust (st kui esimese korraga ei õnnestu, siis võib uuesti proovida, arvesse läheb parim katse).

Demo käigus robotit puudutada ei tohi (kui mõni võistkonnaliikmetest puudutab robotit demo käigus, siis loetakse demo lõpetatuks).

36. Ülesanne – pallini sõitmine

Leida üles väljakult pall, sõita robotiga pallini ning palli ette seisma jääda. Kui pall eemaldatakse, peab robot leidma uue palli, selleni sõitma ning selle ette seisma jääma. Palle on kokku 8 tk.

37. Ülesanne – sponsori hankimine

Hankida projektile sponsorlus:

1. Valida välja mõned firmad, millest sponsorlust küsitakse.
2. Valmistada ette esitlus sponsorile.
3. Küsida firmadelt sponsorlust
4. Kirjeldada ja analüüsida iga sponsori külastust:
 - o kus käisite?
 - o mida küsisite?
 - o mida esitlesite?
 - o kuidas esitlus õnnestus?
 - o mida oleksite võinud teha teisiti?

Vastused vormistada dokumendina korrektse (rtf või pdf) ja esitada moodle keskkonnas.

38. Ülesanne - tulemuste analüüs

Analüüsida oma meeskonna projekti. Analüüsis tuleks täpsemalt vaadata:

- meeskonnatööd
- valitud lahenduse realiseerimist
- projekti tööd

Analüüsi lõppu välja tuua iga teema kohta soovitusel, mida järgmises projektis arvestada. Analüüs vormistada korrektse dokumendina (rtf või pdf) ja esitada keskkonnas Moodle.

39. Ülesanne - aruanne

Kursuse jooksul tehtud projekti kohta tuleb esitada aruanne.

Aruandes kindlasti välja tuua:

1. Lahendusele esitatavad nõuded
2. Lahenduse detailne kirjeldus (lisaks joonised, diagrammid, skeemid jne). Eraldi on vaja välja tuua:
 1. elektroonika kirjeldus - ülesehitus, skeemid jms
 2. mehaanika kirjeldus - koostejoonised koos spetsifikatsiooniga, pildid 3D mudelitest, koostepuu (hõlmab terve koboti koostu) detailidest ja kogustega jms. Valmistustehnoloogia valik materjal - Miks? .Lisaks analüüs valmistatud detailidele (kas vastab või ei miks?).
 3. tarkvara kirjeldus - eraldi lahti kirjeldada madalama ja kõrgema loogika arhitektuur.

Kirjelduste juures ootan ka analüüsi ja mõistlikke põhjendusi, selgitusi lahenduses tehtud valikute kohta. Näiteks peab kindlasti põhjendama mehaanikas materjalide ja töötlusviiside valikuid - kui valiti alumiinium, siis miks nii mitte pleks jms. Tarkvara osas (eriti kõrgema

taseme poolel) selgitada seoseid võistlusülesandega - põhjendada, miks meeskonna poolt valitud tarkvaralahendus aitab hästi võistlusülesannet lahendada. Kui eksisteerib mõistlik lihtsam lahendus ja kasutatud on keerukamat, siis põhjendada, miks. (Kasutada täiendatud versioone ülesande 6 jaoks esitatud elektroonika, mehaanika ja tarkvara arhitektuuridiagrammidest).

Kirjelduste juures eraldi välja tuua:

a) mis on vajalik ülesande täitmiseks - põhiline funktsionaalsus, ilma milleta ülesannet lahendada pole võimalik

b) mis on lahenduse osa, mis annab ülesande täitmiseks paremad võimalused

c) mis on nõ lahenduse "kellad ja viled" - näeb ilus välja, aga ülesande täitmisele kaasa ei aita

4) Roboti eelarve. Eelarves välja tuua eraldi ostetavad ja tellitavad detailid, samamoodi eelarves eraldi välja tuua mehaanika, elektroonika. Tähtsamate detailide puhul põhjendada valikuid.

5) Kirjeldada ära meeskonnatöö korraldus - milliseid vahendeid meeskonnatöö paremaks korraldamiseks kasutasite (skype, tasklist vms?), kuidas jagasite projektijuhi kohustusi, kuidas jagasite ülesanded meeskonna sees, vastutusvaldkonnad jms. Analüüsida meeskonnatööd - mis õnnestus hästi, mis mitte nii hästi, pakkuda välja lahendusi, kuidas teha paremini.

6) määratleda ära ja aruandesse kirjeldada koos projektiplaanidega, eesmärkide ja oodatavate tulemustega projekti kaks osa:

1. projekti osa, mis tehakse/tehti kevadise aine raames

2. projekti osa, mis on plaanitud sügisese aine raames. Kummagi osa kohta tuleb teha projektiplaan koos ajahinnangutega (tundides). Ajahinnangute juures kasutada teadmist, et ühe päev projekti mõttes on 4 h.

7) Projekti 1. osa kohta välja tuua iga inimese kaupa eraldi planeeritud tööde nimekiri (mitte unustada, et tööd ei tehta töö tegemise pärast vaid mingi eesmärgi täitmiseks - seega peab töö juures olema märgitud ära eesmärk ja oodatav tulemus), ära märkida, mis tööd tehtud said, ajahinnangud tundides, tulemused. Tööd tuleb kanda Ganti graafikule. Iga kuu kohta tuleb teha 2 eraldi graafikut - planeeritud tööd ja esialgsed ajahinnangud ja tegelikult tehtud saanud tööd ja realselt kulunud aeg. Graafikute koostamisel on teile abiks meeskonna poolt täidetud tasklist.

8) Projekti analüüs ja tulemused.

* Tuua välja, kui suur osa planeeritust sai valmis ja kui palju jäi projekti töödest tegemata ja eesmärkidest saavutamata. Analüüsida ja pakkuda välja korrektuurid projekti 2. osa plaani.

* Tuua välja iga inimese panus projekti ja analüüsida. Analüüsimisel mitte unustada, et vaatama peaks nii positiivseid kui negatiivseid külgi. Mida teeksite teisiti?

* Tuua välja projekti 2. osa plaani kriitilised ja problemaatilised kohad ja pakkuda lahendusi kriitiliste punktide vältimiseks.

9) Projekti rahastamine. Läbi mõelda ja aruandes kirjeldada plaan sponsorluse hankimiseks - kellelt ja mida küsitakse, mida vastu pakutakse, mida näidatakse. Detailselt! (st lause: "näitame sponsorile valmis asju" ei sobi. Peab lisaks lahti kirjutama, mis need valmis asjad konkreetselt teie projekti puhul on).

10) Oletame, et tuleb firma X ja ütleb teie projekti kohta: "väga äge, tehke mulle ka samasugune lahendus". Koostage hinnapakkumine oma lahenduse kohta firmale X. Mitte unustada, et roboti hind ei ole ainult komponendid vaid ka töö jms. Kirjeldage hinnapakkumist - kui palju on komponendid, kui suur osa tööraha, kui suur osa muud kulud (rent, pinkide kasutus jms).

11) Analüüsige oma hinnapakkumist - kui teie oleksite firma X, siis millega selles pakkumises te ei oleks rahul. Mille eest te ei oleks nõus maksma?

12) Firma X otsustab, et on nõus maksimaalselt projekti eest maksma **6000** EUR. Mida muudate projektis ja pakkumises, et projekt selle raha eest valmis saaks? Millest loobute? Kui palju projekti rahast oleks võimalik teile maksta "kuupalgaks" - arvutage, palju jääb alles raha, kui võtate **6000** EUR-st maha projekti kulud (arvestage sealjuures, et firmale X projekti tehes ei saa projekti võtta asju nii, et "akud, rattad, mootorid, arvuti võtan laborist ja see ei maksa midagi") ja jagate selle meeskonnaliikmete arvuga ja projekti kestvuse kuude ajaga?

Dokumendile lisaks saata selgitus, kus on märgitud iga meeskonnaliikme panus projekti analüüsidokumendi kirjutamisel.

Dokument peab olema korrektselt vormistatud ja esitatud PDF formaadis. Aruanne tuleb üles laadida moodle-sse koos vajalike failidega (elektroonikaskeemid, mehaanika joonised, mudelid jms).

40. Ülesanne – esitlus

Valmistada ette esitlus, milles tutvustada oma projekti. Esitlus peab sisaldama:

1. meeskonna tutvustust ja meeskonnatöö kirjeldust
2. projekti lähteülesande selgitust
3. projektis valitud lahenduse kirjeldust
4. projekti töö kirjeldust
5. projekti töö tulemuste kirjeldust ja analüüsi

Soovituslik on esitlusse planeerida ka valminud roboti demo. Esitlus tuleb vormistada korralike slaididena ja kanda ette aine viimases seminaris.