# Problem definition:

I vores projekt har vi taget kontakt til [www.gamerslounge.dk](http://www.gamerslounge.dk), sammen med dem har vi lavet en aftale om at udarbejde eller hjælpe med at udarbejde et wordpress plugin som omhandler et avanceret turneringssystem. Gamerslounge har bedt om at der skal være flere forskellige turnerings typer tilgængelige for brugere, disse typer er. Single Elimination, Double Elimination, Round Robin, Leagues og Ladder. Den type de har sagt er vigtigst og har højest prioritet er League.

League skal indeholde to hovedområder:

1. Man skal kunne tilmelde Ligaer.
2. Brugere skal kunne arrangere kampe internt.
3. Oprettelse bliver udført af admin
4. Sæson-længde/runde-længde skal kunne vælges under oprettelse
5. Arrangering af kampe skal have en form for sikkerhedsnet, så kampe ikke bliver tabt uden grund. I form af mulighed for at rykke kampe til muligvis andre runder hvis tidspunkter ikke kan findes hvor begge parter kan spille. Men der skal være et maximum antal kampe man kan rykke. Dette maximum har Gamerslounge specificeret som 33% af ens fulde antal kampe, men mindst 3.
6. Der skal være noget notifikations system som fortæller når man har modtaget en ”udfordring”

**Anvendelses område:**

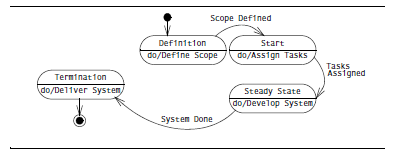
Da dette produkt er til en hjemmeside der bliver brugt af enhver der har lyst til det, og er en form for social hjemmeside er anvendelsesområdet brugerne selv.

**Problem område:**

Dette område består af oprettelse/kontrol af ligaerne og generelt alt admin-arbejdet.

# Initial Software Project Management Plan

For at beskrive vores indledende projektplan tager vi udgangspunkt I følgende figur, da dette er en klassisk arbejdsproces I store træk.



I ovenstående figur ses det at man starter med at beskrive og komme med en definition af ens projekt, hvorefter man uddeler arbejdsroller med henhold til ens kompetencer og ekspertise.

Derefter begynder arbejdet på det givne projekt og til slut afleverer man det færdige produkt, hvis alt går vel. Essensen herved er det som ikke er en del af figuren. Dvs. at alt kommunikationen som ligger imellem alle disse faser. Her i blandt er der fokus på to typer kommunikation, først og fremmest den planlagte kommunikation som hovedsageligt består af aftale møder og mails. Derud over er der den ikke-planlagte kommunikation der består af spontan kontakt i tilfælde af problem opstående eller brug for specifikation.

Der hvor vi differentiere os selv fra denne figur er hovedsagelig uddeling af arbejdsroller. Da vi er en lille gruppe med manglende erfaring, har vi besluttet at den bedste fremgangsmåde ville være at arbejde som en gruppe som kan støtte hinanden og supplere hinanden i svære dele af projektet. Udover dette har vi planlagt at følge den ovenstående figur i grove træk.

Kommunikationsmæssigt har vi at gøre med et lille firma som vi har socialt kendskab til udover projektet. Af denne grund mener vi at der vil komme mere u-planlagt kommunikation end der ellers ville i sådanne projekter. Dog vil vi stadig planlægge møder og eventuelle mail-tidspunkter og skype-opkald.

Med henhold til afslutning af projekt og aflevering af produkt det det eneste der er aftalt med firma at vores arbejdsindsat ophører 5/6-2014.

# Initial Software Architecture

Da firmaet vi arbejder for allerede har en eksisterende hjemmeside og en programmør der arbejder på det samme som os, har vi fået nogle specifikationer omkring krav og kodesprog. Disse krav og specifikationer består blandt andet af at koden til selve webapplikationen som skal kunne kører turneringssystemet skal skrives i PHP/CSS/MySQL, og skrives som et wordpress plugin. Vi er blevet bedt om at koncentrere os om Liga-delen af systemet. Denne del skal dog stadig indeholde profil opsætning, til/af-melding af ligaer, en slags ”matchmaking”-system som foregår manuelt, hvor man tilmelder sig dage og tidspunkter man kan spille og derved tjekker systemet hvilke tider der passer bedst. En nærmere beskrivelse står under afsnit ”Problem Definition” øverst i rapporten.

# Project Agreement Definition

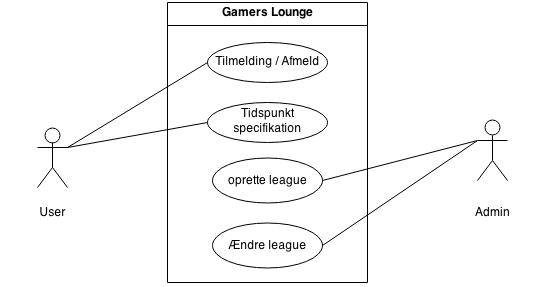
For specifikationerne til turneringssystemet har vi følgende funktionelle og ikke-funktionelle krav, og use-cases.

Funktionelle krav:

* Til/af-melding af ligaer
* Arrangering/rykning af kampe mellem brugere
* Angive ønskede tidspunkter for kamp
* Oprettelse af ligaer (kun for admin)
* Ændring af sæson/runde-længde (kun for admin)
* Sikkerhedsnet for fejl/mangel af aftale om kamp
* Point-system
* Mulighed for at modtage notifikationer om udfordring

Ikke-funktionelle krav:

* Smart design
* Hurtigt og nemt at overskue
* 24/7 adgang
* Systemet skal holde til at mange ligaer skal kunne kører på en gang(holde til overbelastning)
* Mulighed for at kigge på andre ligaer man ikke er tilmeldt
* Systemet bliver vist ordenligt på alle browsere og mobile devices.



**Kompetence Matrix:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Thomas | Kasper | Brian | Michal |
| Java | sekundær-skill/ interesse | sekundær-skill/ interesse | sekundær-skill/ interesse | sekundær-skill/ interesse |
| PHP | interesse | interesse | interesse | interesse |
| HTML | interesse | interesse | interesse | interesse |
| CSS | interesse | interesse | interesse | interesse |

Indtil videre har vi kun gennemført java kurset derfor er denne den eneste som er sat som en faktisk kompetence, hvor resten er interesser. Grunden til at resten er sat som interesse er at vi stadig er meget tidligt i vores uddannelse og derfor ved vi kun basiske aspekter omkring de forskellige sprog, og det gør at vi gerne vil lærer mere og er interesseret i det.