## Användningsfallsmodell

**Aktörer**:

**Gymnastik klubbar**

De olika klubbföreningarna kan besöka webbsidan och registrera sig genom ett formulär som de ska fylla i och skicka iväg.

De kan registrera nya lag med medlemmar samt registrera vilka grenar, ålder och kön de tillhör.

De kan även ta bort lag som de har registrerat.

**Lag ledare**

De vägleder och ger ut information till gymnasterna.

De har samma användningsfall i systemet som gymnastik klubbarna.

**Sekreterare**

Handhåller Tävlingspoängen och räkna ut medelpoängen.

Sekreteraren kan skriva in och ändra poängen i systemet.

**Administratör**

Administratören har behörighet i hela systemet och har i uppgift att uppdatera och redigera scheman, lagen, statistiken samt användarna i systemet.

De bekräftar även nya registreringar som kommer in.

**Huvud domare**

Huvuddomarens uppgift är att informera de andra domarna vilka lag de ska döma samt vilka tider som gäller.

Huvuddomaren kan logga in och ändra i schemat, exempelvis vilka domare som ska döma vilka lag samt tider.

**Domare**

Domarna har till uppgift att tilldela poäng till gymnasterna och ser till att reglerna följs. De har också till uppgift att lämna över poängen till sekreteraren.

**Gymnasterna**

Gymnasterna kan inte registrera sig i systemet utan kan bara gå in och kolla scheman, statistik, namn på domare och namn på lag som ska tävla.

# Användningsfalls beskrivningar

**Registrera användare.**

Registreringen är ett formulär som fylls i på webbsidan och lagras i databasen men innan användaren kan logga in måste administratören godkänna registreringen och dela ut specifika rättigheter till användaren. När administratören har bekräftat registreringen skickas ett mail till användaren att registreringen är genomförd. Vid registrering finns det ett par fält användaren måste fylla I korrekt.

För och efternamn

* Lämpligt reguljärt uttryck för att kola om det är ett riktigt namn.
* Får inte lämnas blankt.

Personnummer

* Kolla så personnumret är valid.

Vilken typ av roll

* En lista med olika roller exempelvis lagledare, sekreterare.
* Ett val är ett krav.

Mail

* Kolla så angiven mail inte redan finns.
* Lämpligt reguljärt uttryck för att kola om det är en riktig mail.

Lösenord

* Minst antal tecken på lösenordet.
* Special tecken eller siffra ska ingå.
* Skriva in lösenordet igen

Namn på klubben

* Får inte lämnas blankt.

Ett robotfilter

* Kollar att robotfiltret är rätt ifyllt.

Ifall användaren hoppa över eller missar ett fält kommer fältet att indikeras med en röd list runt fältet.

Om något av dessa fält Inte fylls i korrekt kommer formuläret inte skickas iväg och en ruta med text kommer informera användaren vart felet är och ligger.

Användaren kan sedan rätta till eventuella fel och skicka iväg formuläret på nytt.

**Beskriv pre- och postvillkor, primära och alternativa flöden**

Previllkor:

* Användaren har gått in på webbsidan och hittat registrerings länken.

Postvillkor:

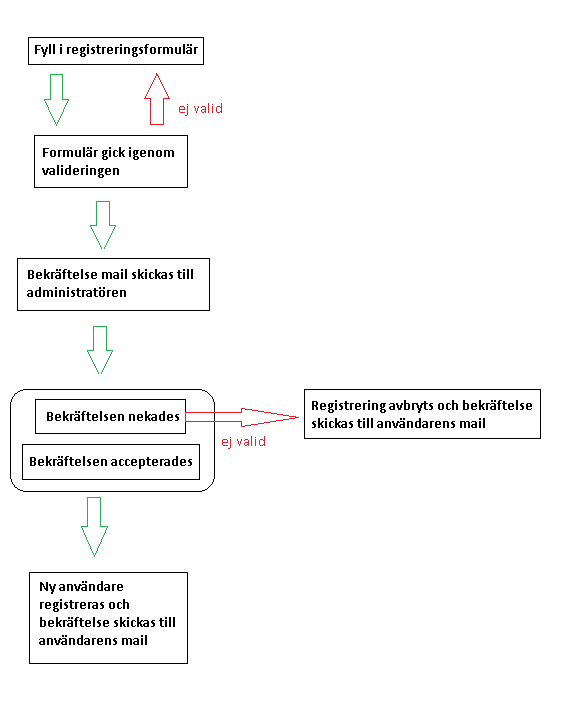
* Registrering av användare är lyckad.
* Alla data som är ifyllt har skickats till databasen.

Primära:

* Bekräftelse på en lyckad registrering (Den perfekta världen)
* Användaren får en bekräftelse till sin mail.

Alternativa flöden:

* Fält som inte är korrekt ifyllt visas genom ett meddelande.
* Fält som är tomma indikeras.



**Registrera poäng.**

Sekreterarna kan logga in på webbsidan och lägga in poäng i systemet.

Genom att gå in på kontrollpanelen visas en lista med alla lagen i bokstavsordning. Listan går även att filtrera genom kön, åldersgrupp och gren, det går även att filtrera bort flera alternativ.

En sökfunktion finns även att tillgå som kan söka efter lag och medlemmar.

Genom att klicka på ett lag i listan genereras en lista upp med lagets namn.

I listan med lagets medlemmar går det att lägga in poäng medlemsvis genom att klicka på ”lägg till poäng” som befinner sig till höger vid medlemens namn.

Vid klick på ”lägg till poäng” knappen genereras ett fält som man kan skriva in poängen i men inte genom text från tangentbordet utan man får klicka sig fram genom pilar.

Genom att klicka sig fram genom pilarna blir det svårare att skriva in fel så som text eller punkter och bindestreck.

Poängen lägger sig bredvid ”lägg till poäng” knappen som kommer ändras till ”ändra poäng” och om sekreteraren väljer att klicka på knappen igen kommer det nuvarande poängen att ändras.

De poäng som har lagts till hos personerna kommer läggas till i lagets totala poäng.

**Beskriv pre och post villkor, primära och alternativa flöden**

Pre villkor:

* Sekreteraren har registrerat sig i systemet
* Administratören har godkänt bekräftelsen
* Sekreteraren har gått in på systemet och navigerats till

registrera poäng sidan

Post villkor:

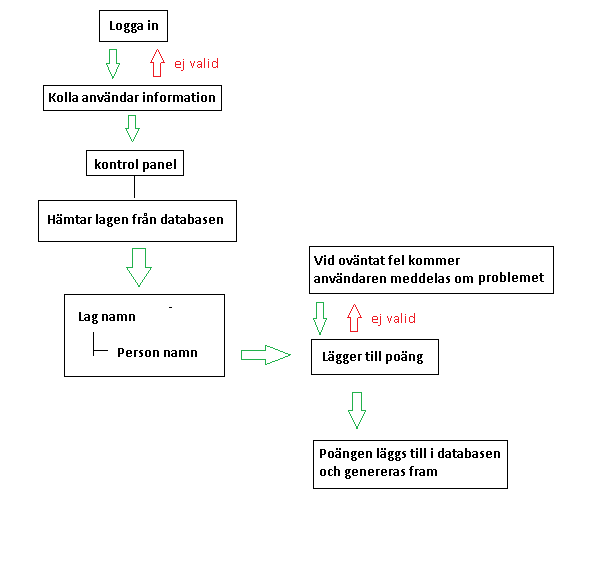
* Registrering av poäng är lyckad
* Alla data som är ifyllt har skickats till databasen

Primära

* Bekräftelse på en lyckad registrering (Den perfekta världen)
* Användaren får en bekräftelse

Alternativa flöden

* Fält som inte är korrekt ifyllt visas genom ett meddelande
* Fält som är tomma indikeras



# Klasser med beskrivningar

**Klass Register**

Klassen är avsedd för aktörer som registrera sig i systemet innehållande fält, metoder, egenskaper och konstruktorer.

**Fält**

**string \_username;**

Innehåller namn på användaren.

**string \_password;**

Innhåller användarens lösenord.

**string \_email;**

Innehåller användarens email.

**Egenskaper**

**String Username**

Ger fältet \_username ett värde av typen string.

**String Password**

Ger fältet \_password ett värde av typen string samt går igenom metoden passValid().

**String Email**

Ger fältet \_email ett värde av typen string samt går igenom metoden emailValid().

**Konstruktorer**

**Register()**

Skapa ett nytt objekt av klassen Register.

Inparametrar är samtliga fält.

**Metoder**

**emailValid()**

Kollar om \_email är en valid mail genom reguljära uttryck.

**passValid()**

Kollar så att \_password innehåller mellan 4 till 8 tecken samt inkludera minst en siffra genom reguljära uttryck.

**Registered()**

Kollar så alla fällt har fått ett värde tilldelat och där med returnera true annars false för att se om registreringen har lyckats.

**Klass Secretary**

Klassen är avsedd för sekreterare som registrera poäng i systemet innehållande fält, metoder, egenskaper och konstruktorer.

**Fält**

**string \_name;**

Innehåller namn på gymnasten.

**double \_score;**

Innehåller gymnastens poäng.

**double \_teamScore;**

Innehåller lagets poäng.

**Egenskaper**

**String Name**

Ger fältet \_name ett värde av typen string.

**double Score**

Ger fältet \_score ett värde av typen double.

**Konstruktorer**

**Secretary()**

Skapa ett nytt objekt av klassen Secretary.

Inparametrar är fälten \_name och \_score

**Metoder**

**ScoreString()**

Returnera fälten \_name samt \_score

**TeamScore()**

Lägger till gymnastens poäng \_score i lagets poäng \_teamScore och returnera \_teamScore

