

## Iteration A

### Planering

Uppgift	Planerad tid	Utförd tid
Planering	1 t	50 m
Analys	1 t	45 m
Implementering av klasser	2 t	1 t 25 min
Testning	3 t	3 t
Reflektion	1 t	50 m

Kommer försöka jobba enligt denna planeringen med arbetet, det vill säga med tidsgränser som ej skall bör överskridas.

### Användningsfall: Inloggningsfunktion

Detta användningsfallet kommer beskriva hur inloggning i systemet kommer fungera.

#### Aktörer

- Användare
- Lag
- Gymnaster
- Domare/juryn
- Klubbbar

#### Pre-villkor

Användarna är registrerade i systemet och har tillgång till sina personliga inloggningsuppgifter.

Användaren får verifikations mail vid registrering av sin profil i systemet.

Användaren som skall registrerar sig behöver ha tillgång till sina personuppgifter eller om det är en klubb som skall registrera sig så behöver de ha tillgång till sitt organisationsnummer.

#### Post-villkor

Användaren är inloggad i systemet.

Användaren får mail angående om någon har skrivit fel lösenord eller användarnamn. Vid detta spärras även profilen och användaren måste då kontakta supporten.

## Primära flöden

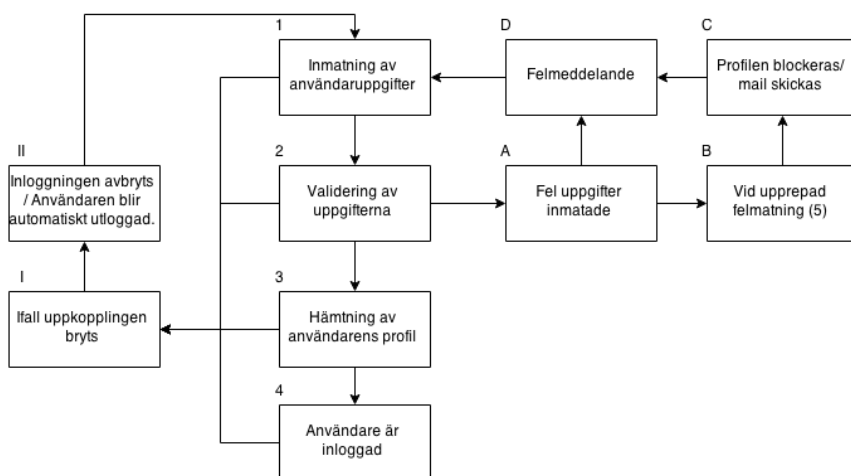
1. Användaren matar in sina inloggningsuppgifter.
2. Systemet validerar uppgifterna.
3. Systemet hämtar användarens profil.
4. Användaren är nu inloggad i systemet.

## Alternativa flöden

- A. Fel uppgifter inmatade.
- B. Vid upprepad felinmatning (5gånger).
- C. Profilen blockeras/Mail skickas.
- D. Felmeddelande.

## Exceptionella flöden

- I. Ifall uppkopplingen skulle brytas av okänd anledning.
- II. Inloggningen avbryts / användaren blir automatisk utloggad.



## Analys

### Klasser

Efter analys av flödena som jag har planerat så planerar jag att ha:

#### Klassen User:

I klassen User kommer jag sätta en bestämd längd på användarens lösenord och användarnamn. Kommer även införa en regex för användarmailen.

- Bestämd längd på lösenord och användarnamn
- Regex
- Email

#### Klassen Validate:

I klassen Validate kommer jag validera lösenordet och användarnamnet som användaren matar in vid inloggning. Om användarnamnet/lösenordet upprepas fel fem gånger så kommer även det ett felmeddelande och vid det femte försöket så kommer det skickas ett mail till användaren och man kommer ej kunna logga in.

- Användarnamns validering
- Felmeddelande
- Userblock

Metod	Testfall	Förväntat resultat	Resultat
TestPasswordCorrect	Godkänd data matas in	metoden returnerar true	Testet lyckades
TestPasswordIncorrect	Icke godkänd data matas in	Metoden returnerar false	Testet lyckades
TestUsernameCorrect	Godkänd data matas in	Metoden returnerar true	Testet lyckades
testUsernameIncorrect	Icke godkänd data matas in	Metoden returnerar false	Testet lyckades
TestEmailPass	Godkänd data matas in	Metoden returnerar true	Testet lyckades
TestEmailFail	Icke godkänd data matas in	Metoden returnerar false	Testet lyckades
TestValidate	Godkänd data matas in	Metoden returnerar true	Testet lyckades
WrongTestValidate	Icke godkänd data matas in	Metoden returnerar false	Testet lyckades

## Reflektion

Jag valde att fortsätta bygga på min föregående iteration och lägga till fler test. Detta gjorde jag för att jag kände att jag kunde förbättra iterationen och göra den lite större och ändå få med det jag ville ha för att få mina iterationer att kunna var i samma projekt.

Tidsmässigt så gick denna iterationen för bra för att jag ska vara nöjd. Skulle nog ha kunnat lägga till mer tester och mer funktioner men eftersom jag vill hinna med allt så känner jag att jag vill gå vidare på nästa iteration så fort som möjligt ifall jag skulle få problem där. Om detta skulle hända så skulle jag inte kunna få problem med nästa iteration och då skulle det bildas en kedjereaktion igenom alla iterationer.