|  |
| --- |
|  |



Designdokumentation

Grupp 5

Version 1.3 2015-05-22

I kursen Systemutveckling och projekt, DA336A på Malmö Högskola.

Emma Persson Wik 19950306 Informationsarkitekt

Mimmi Castmo 19930528 Informationsarkitekt

Marielle Lucana 19940930 Systemutvecklare

Pernilla Stagnebo 19871216 Informationsarkitekt

Simon Persson 19900401 Informationsarkitekt

# Revisionshistorik

|  |  |  |  |
| --- | --- | --- | --- |
| **Namn** | **Datum** | **Beskrivning** | **Version** |
| Mimmi Castmo | 24032015 | Skapat en första struktur | 0.1 |
| Simon Persson | 27032015 | Lagt in diagram, bilder m.m | 0.2 |
| Simon Persson | 05042015 | Use case m.m | 0.3 |
| Mimmi Castmo | 07042015 | Ändrat rubriker | 0.4 |
| Emma Persson Wik | 12042015 | Lagt in loggan | 0.5 |
| Mimmi Castmo | 05052015 | Samma struktur, lagt in sidhuvud o fot | 0.6 |
| Pernilla Stagnebo | 07052015 | Uppdaterat försättsblad och sidfot. ER-diagram | 0.7 |
| Simon Persson | 12052015 | Uppdaterat Use-case och lagt till skisser | 0.8 |
| Simon Persson | 16052015 | Uppdaterat Use-case, DFD & lagt till fil diagram, navigationsdiagram, kontext beskrivningar. | 0.9 |
| Simon Persson | 18052015 | Lagt till systemdiagram & beskrivning | 1.0 |
| Simon Persson | 20052015 | Dokumentbeskrivning/struktur | 1.1 |
| Marielle Lucana | 20052015 | klassdiagram | 1.2 |
| Simon Persson | 21052015 | Beskrivningar | 1.3 |
| Simon Persson | 22052015 | Syfte, avslutande del. Struktur på nytt. Innehållsförteckning. | 1.4 |
| Marielle Lucana | 24052015 | Klassdiagram, la till och ändrade | 1.5 |
|  |  |  |  |
|  |  |  |  |

Innehållsförteckning

[Revisionshistorik 1](#_Toc420064356)

[1 Inledning 3](#_Toc420064357)

[1. 1 Dokumentstruktur 3](#_Toc420064358)

[1.2 Dokumentets omfattning 3](#_Toc420064359)

[1.3 Syfte 3](#_Toc420064360)

[2 Sagolands övergripande arkitektur 5](#_Toc420064361)

[2.1 Kontextdiagram 5](#_Toc420064362)

[2.2 System-diagram 8](#_Toc420064363)

[2.3 ER-diagram och tabeller 10](#_Toc420064364)

[2.4 Klassdiagram 11](#_Toc420064365)

[2.5 Use Case-diagram 12](#_Toc420064366)

[2.6 Filstruktur Sagoland 19](#_Toc420064367)

[2.6 Navigationsdiagram 20](#_Toc420064368)

[3 Design 21](#_Toc420064369)

[3.1 Design/GUI 21](#_Toc420064370)

[3.2 Skisser 21](#_Toc420064371)

[3.2 Wireframes 23](#_Toc420064372)

[3.3 Slutgiltig design 25](#_Toc420064373)

# 1 Inledning

## 1. 1 Dokumentstruktur

Dokumentet är strukturerat på följande sätt, det börjar med en inblick över produktens övergripande arkitektur, här redovisas produktens insida och hur det är utformat med hjälp av diagram och beskrivningar av input/output samt hur varje aktör står i relation till varandra. Med aktör syftar vi på användaren, webbplatsen och databasen.  
Efter detta kommer redovisningen för produktens utsida, design och GUI är i fokus. Med hjälp av skisser och wireframes beskriver vi hur designfasen har gått till.   
Dokumenten avslutas med en redovisning för produktens slutgiltiga design. Även detta är redovisat i form av bilder från produktens aktuella utseende.

## 1.2 Dokumentets omfattning

Som det står i dokumentstrukturen omfattar detta dokument tre områden. Inblick i produktens insida och hur relationerna står sig mellan olika aktörer. Detta är redovisat med kontextdiagram, systemdiagram, DFD, ER-diagram och slutligen ett klassdiagram över produktens inloggningssystem. Varje diagram är uppföljt av en beskrivning där vi förklarar diagramen noggrant. En genomgående rödtråd i dokumentets beskrivning av produktens insida är input och output. Då vår produkt är en hemsida har vi valt att fokusera mycket på detta för att kunna redovisa hur vår produkt reagerar på användarens input och vilken relation detta skapar mellan produkten och användaren.

Sedan har vi det mer ytliga i dokumenten som fokuserar på den grafiska utformningen av produkten. Genom användning av skisser, wireframes och bilder vill vi så tydligt som möjligt beskriva vår designprocess. Här följs även varje del i detta kapitel utav en beskrivning som förtydligar våra tankar och beslut som vi har tagit. Slutligen avslutar vi dokumentet med att redovisa för den slutgiltiga designen för produkten.

Vi ville med denna struktur skapa en rödtråd genom hela utvecklingen från skiss till den slutgiltiga designen.

## 1.3 Syfte

Syftet med detta designdokument är att redovisa för hur produkten Sagoland är utformat och hur dess olika delar står i relation med varandra. Även ge en inblick i produktens inredelar och hur produkten är utformad för att tillfredsställa våra målgruppers behov, utifrån ett designperspektiv.

# 2 Sagolands övergripande arkitektur

## 2.1 Kontextdiagram

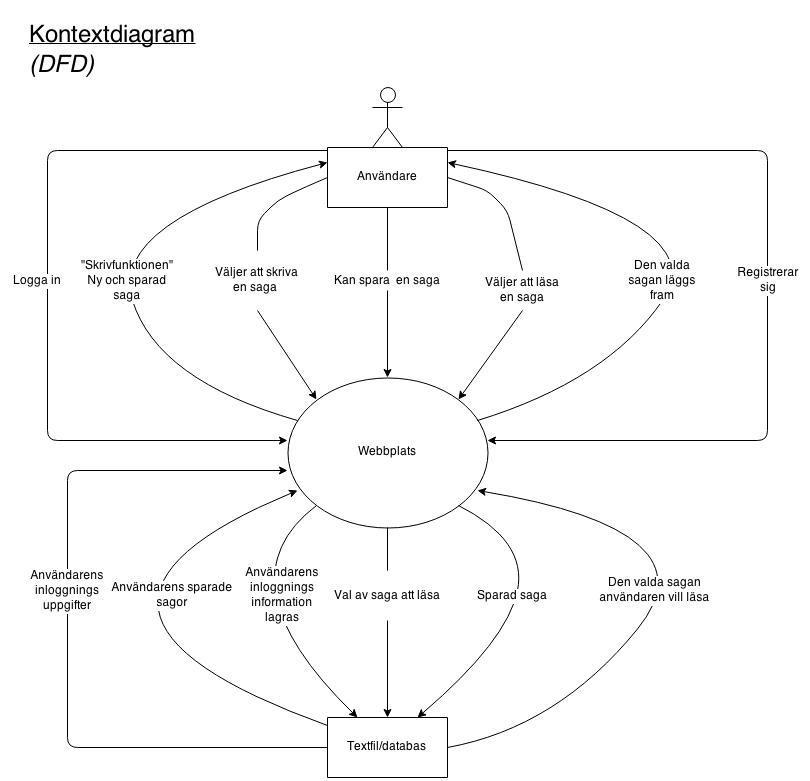


Fig. 2.1

**Beskrivning kontextdiagram**

I Sagolands kontextdiagram finns tre huvudaktörer och i fig 2.1 beskrivs relationen mellan dessa tre huvudaktörer. Pilarna i fig 2.1 demostrerar vilka input samt outputs som utförs i dessa relationer mellan varje aktör. För att skapa en tydligare bild av hur sambanden uppstår följer en detaljerad beskrivning av varje aktörs utförande för att skapa inputs och output.

|  |
| --- |
| **Input: Användaren → Webbplats** |
| *Väljer att läsa en saga:*   1. Användaren går till ”Läsa Saga” sidan. 2. Användaren väljer en saga. 3. Användaren klickar igenom sagan   *Registrerar:*   1. Användaren går till ”Logga in” sidan. 2. Användaren fyller in den nödvändiga informationen i formuläret. 3. Användaren skickar informationen till databasen.   *Logga in:*   1. Användaren går till ”Logga in” sidan. 2. Användaren fyller i sitt användarnamn/lösenord. 3. Användaren klickar på ”logga in”.   *Väljer att skriva en saga:*   1. Användaren går till ”Skriva Saga” sidan. 2. Användaren skriver en egen saga i textfältet. 3. Användaren ger sin saga en titel. 4. Användaren ändrar färg och textstil på sagan. 5. Användaren lägger till bilder i sin saga.   *Väljer att spara sin saga:*   1. Användaren vill spara sin färdiga saga. 2. Användaren klickar på ”Spara” knappen. 3. Användaren namnger sin Saga. |
| **Input: Webbplats → Textfil/Databas** |
| *Användarens inloggningsinformation:*   1. Användarens inloggningsinformation skickas till databasen. 2. Databasen registrerar användarens information.   *Val av saga:*   1. Användarens val av saga skickas till databasen.   *Sparad saga:*   1. Användaren vill fortsätta på en sparad saga. 2. Val av sparad saga skickas till databasen. |
| **Output: Textfil/Databas → Webbplats** |
| *Användarens inloggningsinformation:*   1. Databasen kontrollerar att informationen är korrekt. 2. Databasen godkänner användaren.   *Användarens sparade saga:*   1. Den lagrade textfilen hämtas. 2. Textfilen skickas till skriv sidan.   *Den valda sagan användaren vill läsa:*   1. Den valda sagan identifieras. 2. Sagan hämtas. 3. Sagan skickas till Läsa sidan. |
| **Output: Webbplats → Användare** |
| *Den valda sagan läggs fram:*   1. Läsa sidan visar upp den valda sagan för användaren. 2. Användaren kan klicka igenom sagan. 3. Sidan erbjuder användaren att interagera med den valda sagan.   *Skrivfunktionen, ny och sparad saga:*   1. Användaren kan på skriv sidan ta del av skrivfunktionen. 2. Användaren kan ändra färg på text. 3. Användaren kan ändra stil på text. 4. Användaren kan lägga till bilder. 5. Användaren kan fortsätta på en sparad saga. 6. Den sparade sagan läggs fram i text rutan. |

# 

## 2.2 System-diagram

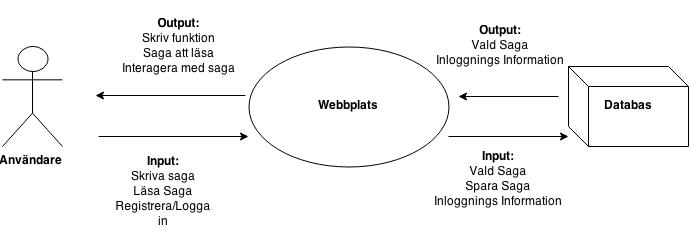


Fig. 2.2

**Beskrivning system-diagram**

Ett mer övergripande diagram över Sagolands system. Användaren som är ”*klienten*” arbetar mot webbplatsen som motsvarar en ”*server*”, webbplatsen kommunicerar sedan med en databas som hanterar inloggningssystem och laggring av sagor. I fig 2.2 är kommunikationen mellan klienten, servern och databasen är utsatt i form utav pilar som visar vart kommunikationen kommer från och vad som skickas tillbaka. För att skapa en tydligare bild av denna kommunikation följer en detaljerad beskrivning av hur klienten, servern och databasen skapar input och output genom kommunikation.

**Användare/Klient**

Input

* Användaren väljer vad den vill göra på hemsidan genom att aktivt välja Läsa eller Skriva en saga. Om användaren enbart vill läsa en saga krävs det att användaren väljer en saga. Valet av saga skickas sedan till databasen som laddar sagan.
* Även kunna göra aktiva val för sagans utfall och ändra namn på dess karaktärer.
* För att kunna skriva en saga krävs det att användaren registrerar sig på hemsidan. För att göra detta behöver användaren fylla i ett formulär på hemsidan som sedan skickas iväg via hemsidan till databasen där informationen lagras.
* Nu kan användaren använda sig utav skrivfunktionen på hemsidan. Användaren kan genom denna funktion skriva sin text i en textruta. Användaren kan även ändra färg och stil på sin text, detta är val som användaren måste aktivt göra.

**Webbplats/Server**

Input

* Webbplatsen informerar databasen om vilken saga användaren vill läsa eller vilken sparad saga den vill att databasen ska ladda.
* Även användarens inloggningsinformation skickas vidare till databasen.

Output

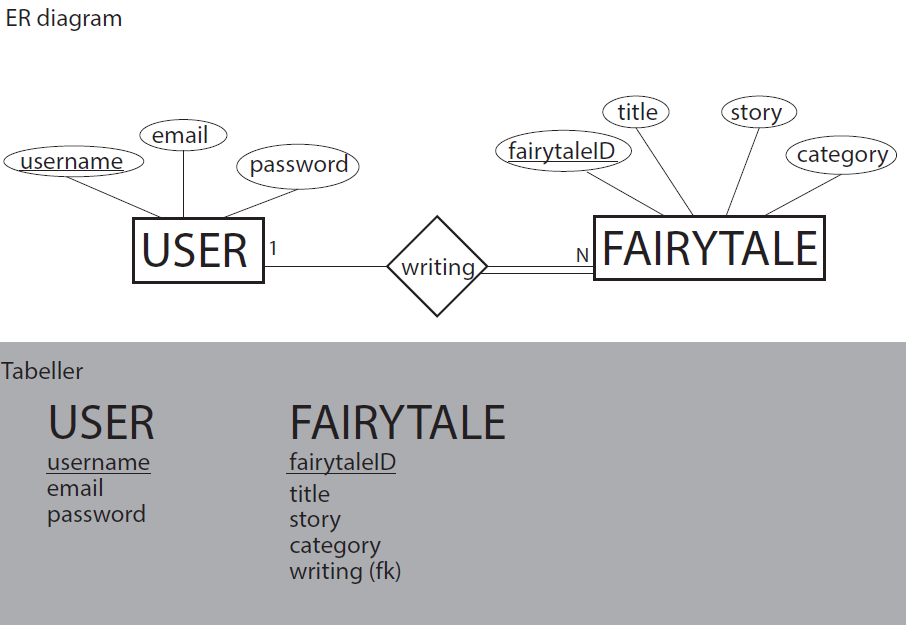
* Webbplatsen ger användaren möjlighet att kunna läsa den valda sagan som skickas från databasen.
* Webbplatsen ger användaren möjlighet att kunna använda sig utav skrivfunktionen samt att visa den sparade sagan som användaren väljer att fortsätta på.
* Även genom att visa resultatet av användarens val när det kommer till att interagera med sagorna i Läsa delen.

**Databas**

Output

* Databasen hämtar den saga som användaren vill läsa.
* Databasen hämtar den saga som användaren vill fortsätta att skriva på.
* Databasen kontrollerar att användarens inloggningsuppgifter stämmer överens med den lagrade informationen.

## 2.3 ER-diagram och tabeller



**Beskrivning ER-diagram**

Detta är en övergripande bild av hur vi har strukturerat upp vår databas. Vi har valt att fokusera mest på inloggningssystemet, som är ”USER” delen. Här har vi bestämt att användaren ska ha ett användarnamn som är nyckeln och därför unik sedan ska användarens mailadress och lösenord lagras i vår databas.

## 2.4 Klassdiagram

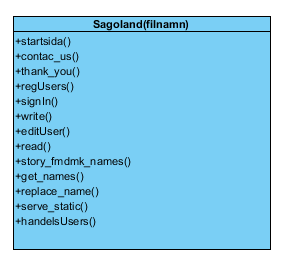


Fig. 2.4

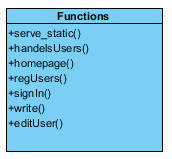


Fig. 2.5

**Beskrivning klassdiagram**

Klassdiagrammet ”Sagoland” visar hur strukturen i klassen ser ut samt vilka metoder som är publika och privata.

Klassdiagrammet ”Function” visar hur strukturen i klassen ”Function” ser ut samt vilka metoder som finns och om de är privata eller publika. Detta klassdiagram är utformat efter Sagolands inloggningssystem.

## 2.5 Use Case-diagram

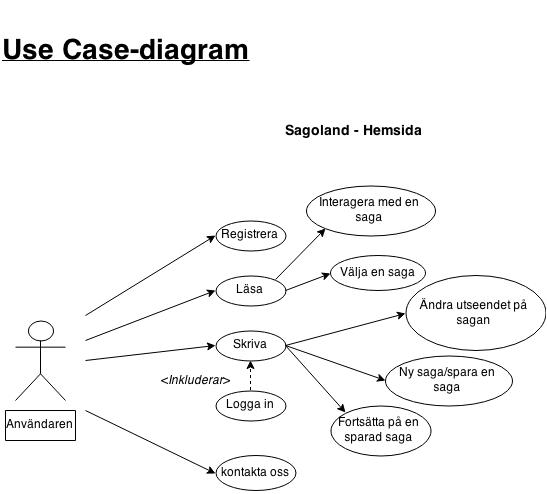


Fig. 2.6

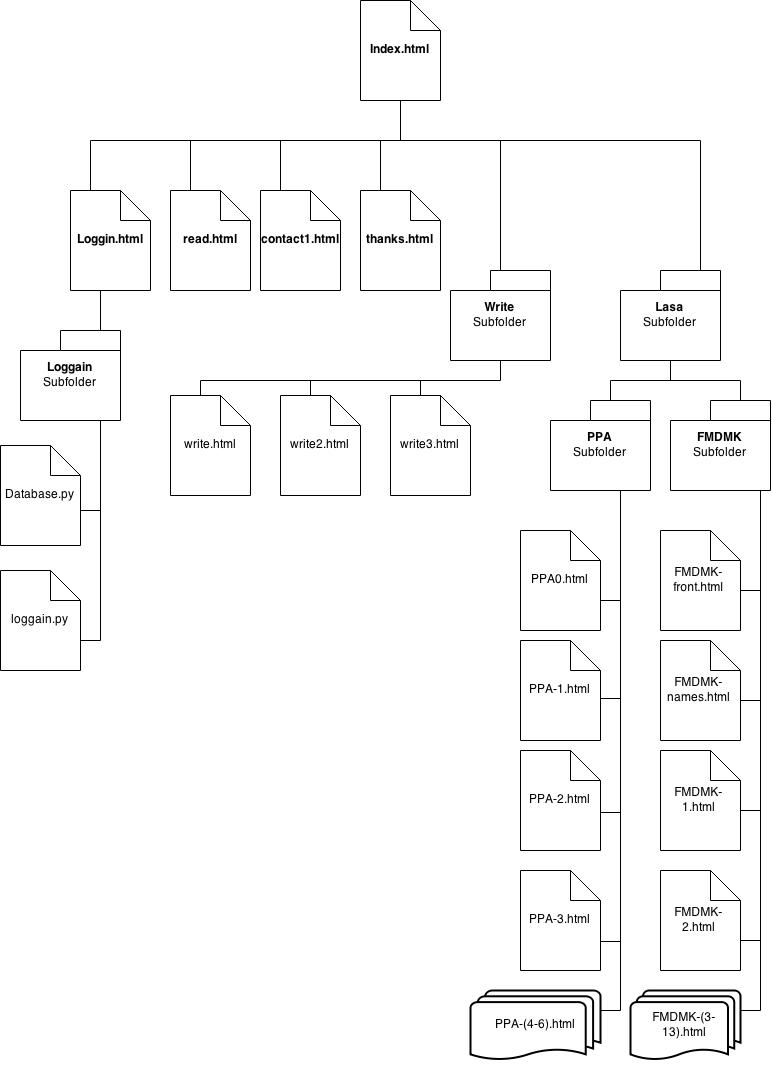
**Beskrivning Use-case**

Syftet med ett Use-case diagram är att visa hur användaren ska interagera med vår produkt Sagoland. I fig 2.6 får vi en övergripande bild utav detta. För att förtydliga diagramet har vi beskrivit varje scenario med en tydlig beskrivning för varje steg i processen. Användaren i fig 2.6 är enbart en klient och ska därför inte förväxlas med administratör.

Scenario Beskrivning

|  |  |
| --- | --- |
| **Registrering** | *För att kunna använda sig utav hemsidans skrivfunktion så måste användarna först registrera sig.*  **Förutsättningar**: En ny användare vill registrera sig.  **Huvudflödet**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram inloggningssidan. 4. Användaren klickar på ”registrera ny användare”. 5. Systemet tar fram registreringssidan. 6. Användaren fyller i all den nödvändiga informationen som krävs. 7. Användaren klickar sedan på skicka knappen. 8. Systemet kontrollerar att den nödvändiga informationen är ifylld. 9. Systemet registrerar användarens uppgifter. 10. Användaren är nu inloggad. 11. Systemet tar fram indexsidan. |
| **Inloggning** | *För att kunna använda hemsidans skrivfunktion så måste användarna logga in på hemsidan.*  **Förutsättningar**: Användaren finns registrerad i databasen och uppger rätt lösenord/användarnamn.  **Huvudflödet**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram inloggningssidan. 4. Användaren fyller i sitt användarnamn och sitt lösenord. 5. Användaren klickar på ”logga in” knappen. 6. Systemet kontrollerar att inloggningsuppgifterna stämmer mot kontouppgifterna i databasen. 7. Systemet tar fram indexsidan.   **Förutsättningar**: Användaren uppger fel lösenord/användarnamn.  **Huvudflödet**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram inloggningssidan. 4. Användaren fyller i användarnamn och lösenord. 5. Systemet kontrollerar att inloggningsuppgifterna stämmer mot kontouppgifterna i databasen. 6. Användarens inloggnings information stämmer inte. 7. Systemet informerar användaren att användarnamnet/lösenordet är felaktigt. 8. Användaren får ett nytt försök. |
| **Läsa:** Välj saga. | *Hemsidan erbjuder en läsfunktion. Här kan användarna välja mellan olika sagor.*  **Förutsättningar**: Användaren vill läsa en saga.  **Huvudflöde**:   1. Användaren klickar på ”Läs Saga” ikonen på indexsidan. 2. Systemet tar fram läs sidan. 3. Användaren klickar på en av böckernas titel. 4. Systemet tar fram den valda sagan från databasen. 5. Användaren vänder blad i sagan genom att klicka på den sida i sagan som användaren vill vända. 6. När användaren har läst färdigt sagan klickar hen på den sista sidan i sagan. 7. Systemet tar tillbaka användaren till Läsas startsida. |
| **Läsa**: *Interagera med en saga.* | *Hemsidan har en del sagor som användarna kan interagera med.*  **Förutsättningar**: Användaren vill ändra namn på en eller flera karaktärer.  **Huvudflöde**:   1. Användaren klickar på ”Läs Saga” ikonen på indexsidan. 2. Systemet tar fram läs sidan. 3. Användaren klickar på *”Emma med de magiska kritorna”*. 4. Systemet tar fram den valda sagan från databasen. 5. Användaren vänder blad i sagan genom att klicka på den sida i sagan som användaren vill vända. 6. Användaren får en möjlighet att ändra namn på karaktärerna i sagan innan den börjar. 7. Användaren väljer att ta bort ”Emma” i inputfältet och ersätter med ett eget namn. 8. Användaren väljer att ta bort ”Kattis” i inputfältet och ersätter med ett eget namn. 9. Användaren klickar på ”Okej!” Knappen. 10. Systemet ändrar namnet på karaktärerna i sagan. 11. Användaren kan nu läsa sagan med andra namn på karaktärerna. 12. När användaren har läst färdigt sagan klickar hen på den sista sidan i sagan. 13. Systemet tar tillbaka användaren till Läsas startsida.   **Förutsättningar**: Användaren vill välja utfallet av en händelse i sagan.  **Huvudflöde**:   1. Användaren klickar på ”Läs Saga” ikonen på indexsidan. 2. Systemet tar fram läs sidan. 3. Användaren klickar på *”Emma med de magiska kritorna”*. 4. Systemet tar fram den valda sagan från databasen. 5. Användaren vänder blad i sagan genom att klicka på den sida i sagan som användaren vill vända. 6. Användaren får en möjlighet att ändra namn på karaktärerna i sagan innan den börjar. 7. Användaren klickar på ”Okej!” Knappen. 8. Systemet ändrar inte namnet på karaktärerna. 9. Användaren får efter fem sidor möjlighet att välja ett utfall i sagan. 10. Användaren klickar på ett av alternativen. 11. Systemet laddar användarens val av utfall. 12. Användaren vänder blad i sagan genom att klicka på den sida i sagan som användaren vill vända. 13. När användaren har läst färdigt sagan klickar hen på den sista sidan i sagan. 14. Systemet tar tillbaka användaren till Läsas startsida. |
| **Skriva** | *Hemsidan erbjuder användarna att skriva en egen saga.*  **Förutsättningar**: Användaren är inloggad på hemsidan och vill skriva en saga och avsluta med att spara sagan.  **Huvudflöde**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är inloggad. 3. Systemet tar fram ”write” sidan. 4. Användaren klickar på ”Ny saga”. 5. Systemet tar fram ”write2” sidan. 6. Användaren börjar skriva sin saga i textfältet. 7. Användaren klickar på spara knappen. 8. Användaren ger texten ett namn. 9. Systemet sparar användarens text som en textfil. 10. Användaren väljer vart sagan ska sparas.   **Alternativt** **flöde**:   1. Användaren klickar på ”Skriv Saga” ikonen i navigationsfältet. 2. Systemet kontrollerar att användaren är inloggad. 3. Systemet tar fram ”write” sidan. 4. Användaren klickar på ”Ny saga” för att börja skriva. 5. Systemet tar fram ”write2” sidan. 6. Användaren börjar skriva sin saga i textfältet. 7. Användaren klickar på spara knappen. 8. Användaren ger texten ett namn. 9. Systemet sparar användarens text som en textfil. 10. Användaren väljer vart sagan ska sparas.   **Förutsättningar**: Användarna är inte registrerad/inloggad och vill skriva en saga och avsluta med att spara sagan.  Huvudflöde:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är inloggad. 3. Systemet tar fram inloggnings/registrerings sidan. 4. (Inloggningsflöde/registreringsflöde). 5. Systemet tar fram ”write” sidan. 6. Användaren klickar på ”Ny saga” för att börja skriva. 7. Systemet tar fram ”write2” sidan. 8. Användaren börjar skriva sin saga i textfältet. 9. Användaren klickar på spara knappen. 10. Användaren ger texten ett namn. 11. Systemet sparar användarens text som en textfil. 12. Användaren väljer vart sagan ska sparas. |
| **Skriva:** *Fortsätta att skriva på sparad saga.* | *Hemsidan ger användarna möjlighet att spara sin saga för att fortsätta på den vid ett annat tillfälle.*  **Förutsättningar**: Användaren är en registrerad/inloggad användare på hemsidan och har sparat en saga.  **Huvudflöde**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram ”write” sidan. 4. Användaren klickar på ”Fortsätt på en gammal saga”. 5. Systemet tar fram ”write2”. 6. Användaren klickar på ”Ladda Saga” 7. Användaren klickar på den saga hen vill fortsätta på. 8. Systemet laddar användarens text i textfältet. 9. Användaren kan nu fortsätta att skriva på sin saga. |
| **Skriva:** Ändra utseende på sagan | *Hemsidan erbjuder användarna olika sätt att förändra utseendet på sin saga. Användarna kan välja mellan olika färger och stilar på texten samt lägga till bilder för att göra sagan mer personlig.*  **Förutsättningar**: Användaren är registrerad/inloggad och vill ändra färg på sin text.  **Huvudflöde**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram ”write” sidan. 4. Användaren klickar på ”Ny saga” för att börja skriva. 5. Systemet tar fram ”write2” sidan. 6. Användaren börjar skriva sin saga i textfältet. 7. Användaren klickar på ”Ändra färg på din text”. 8. Systemet tar fram de olika färgalternativen. 9. Användaren klickar på den färg hen vill att texten ska få. 10. Systemet ändrar färgen på texten. 11. Användaren kan fortsätta att skriva på sin saga.   **Förutsättningar**: Användaren är registrerad/inloggad och vill ändra utseendet på sin text.  **Huvudflöde**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram ”write” sidan. 4. Användaren klickar på ”Ny saga” för att börja skriva. 5. Systemet tar fram ”write2” sidan. 6. Användaren börjar skriva sin saga i textfältet. 7. Användaren klickar på ”Ändra utseende på din text”. 8. Systemet tar fram de olika textstilarna. 9. Användaren klickar på den textstil hen vill att texten ska få. 10. Systemet ändrar textstilen på texten. 11. Användaren kan fortsätta att skriva på sin saga.   **Förutsättningar**: Användaren är registrerad/inloggad och vill lägga till bilder och ge sin saga en egen titel.  **Huvudflöde**:   1. Användaren klickar på ”Skriv Saga” ikonen på indexsidan. 2. Systemet kontrollerar att användaren är registrerad/inloggad. 3. Systemet tar fram ”write” sidan. 4. Användaren klickar på ”Ny saga” för att börja skriva. 5. Systemet tar fram ”write2” sidan. 6. Användaren klickar på ”Vad heter din saga?”. 7. Användaren skriver in en egen titel på sagan. 8. Systemet ändrar titeln på sagan. 9. Användaren klickar på ”Bilder till sagan”. 10. Systemet tar fram bilderna. 11. Användaren dubbelklickar på den utvalda bilden. 12. Systemet skickar bilden till sagan. 13. Användaren kan positionera bilden genom att dra i bilden. 14. Systemet tillåter användaren att ändra position på bilden. |
| **Kontakt** | *Hemsidan har en kontakta oss sida där användarna kan komma i kontakt med ägarna av hemsidan, gällande diverse frågor.*  **Förutsättningar**: Användaren har frågor om hemsidan.  **Huvudflöde**:   1. Användaren klickar på kontakta oss fliken i navigationsfältet. 2. Systemet tar fram ”contact1” sidan. 3. Användaren läser under rubriken FAQ för att få svar på en rad olika frågor. 4. Användaren fick svar på sin fråga.   **Förutsättningar**: Användaren hittar inte svaret på sin fråga under FAQ.  **Huvudflöde**:   1. Användaren klickar på kontakta oss fliken i navigationsfältet. 2. Systemet tar fram ”contact1” sidan. 3. Användaren läser under rubriken FAQ för att få svar på en rad olika frågor. 4. Användaren hittar inte svaret under FAQ. 5. Användaren fyller i sin mailadress i inputboxen för ”E-postadress”. 6. Användaren skriver sin fråga i textrutan. 7. Användaren klickar på ”skicka meddelandet” 8. Systemet registrerar användarens fråga. 9. Systemet informerar användaren ”*Tack för att du har skickat ett meddelande till oss, vi ska försöka att svara så snabbt som möjligt!”* |

## 2.6 Filstruktur Sagoland



**Beskrivning filstruktur**

Detta är Sagolands filsstruktur, här kan vi se hur filerna står i relation till varandra. Loggain, Write och Lasa är mappar som i sin tur innehåller mappar/filer. Att write och lasa ligger i separata mappar är för att skapa en struktur som underlättar projektets arbete, dessa två funktioner är produktens största och innehåller därför flest filer. Loggain har innehållet av databasen och inloggningssystemet. Write innehåller skrivfunktionen och Lasa innehåller alla sagor som användaren kan läsa. PPA och FMDMK avslutas med symboler som indikerar att det finns fler filer i den mappen.

## 2.7 Navigationsdiagram

**Beskrivning utav navigationsdiagram**

Här kan vi se hur användaren kan navigera sig runt på Sagoland. Pilarna indikerar åt vilket håll användaren kan navigera åt. För att kunna komma åt skrivfunktion som ligger på ”Write” sidan, måste användaren först registrera/logga in för att kunna ta del av denna funktion.

PPA och FMDMK är sagor som innehåller flera html sidor och är därför utformade på det sättet för att symbolisera flera html sidor.

# 3 Design

## 3.1 Design/GUI

Sagolands huvudmålgrupp är barn och därför måste designen vara anpassad efter denna målgrupp. I vår slutgiltiga design fokuserade vi mycket på att skapa en logisk navigationsväg igenom hela produkten genom att förstora och centrera navigationsvalen hoppas vi att vår målgrupp enkelt ska förstå hur de ska ta sig runt och interagera med Sagoland. Vi har även riktat in oss på att vara pedagogiska i en viss utsträckning, genom att ha stora tydliga bostäver och en färgrik hemsida.

Vår andra målgrupp är föräldrar och med det i tanken vid skapandet av designen vill vi även att vår produkt ska ge ett professionellt och seriös uttryck. Detta har vi främst fokuserat på genom att ha en mindre navigations bar som är stilren samt skapandet av ett inloggningssystem. Med detta hoppas vi att föräldrar ska kunna känna att vår produkt är något som de kan känna sig trygga med och förhoppningsviss rekommendera vidare till andra föräldrar.

## 3.2 Skisser

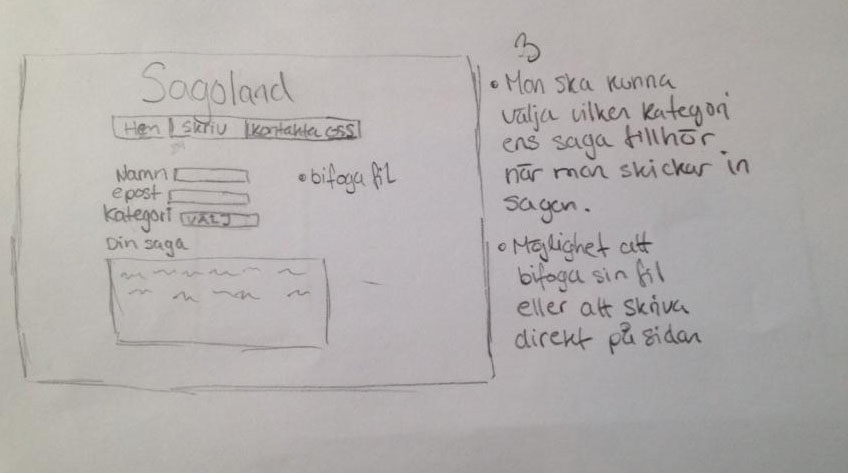
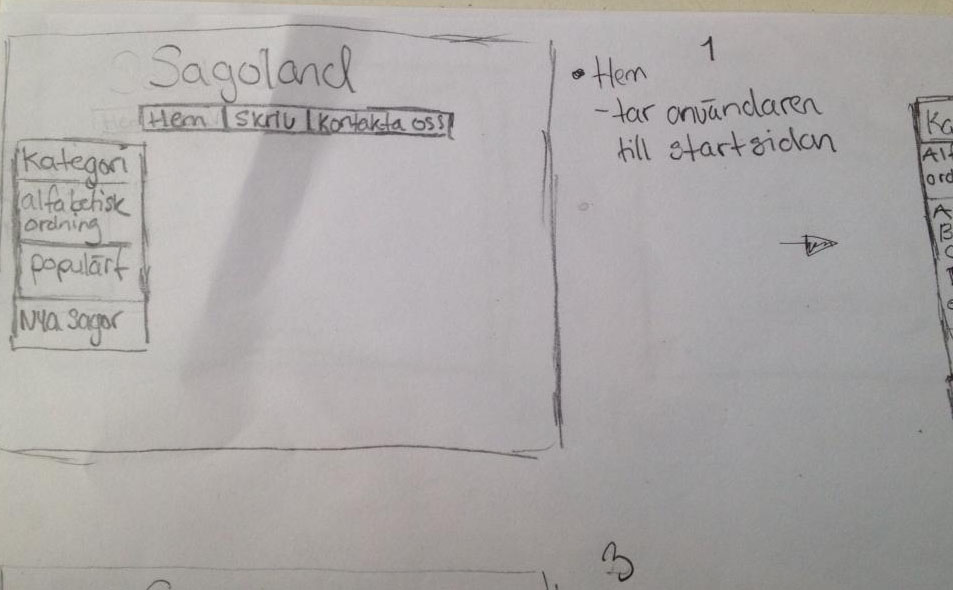
Projektets första designfas var utformningen av ett par simpla skisser som är gjorda för hand. Syftet med dessa skisser var för att skapa en gemensam uppfattning om hur designen skulle kunna se ut. I detta stadie var det mycket fokus på att designa för barn. Genom att använda mycket färger och blommor var målet att skapa en glad och avslappnad miljö (Se fig 3.1–3.4 ). Vi avslutade detta stadiet med att skissa lite runt utseendet på ett kategorisystem och inloggningssystem (Se fig. 3.5 – 3.6).



*Fig. 3.1 Skiss indexsida* *Fig. 3.2 Skiss kategorisida*



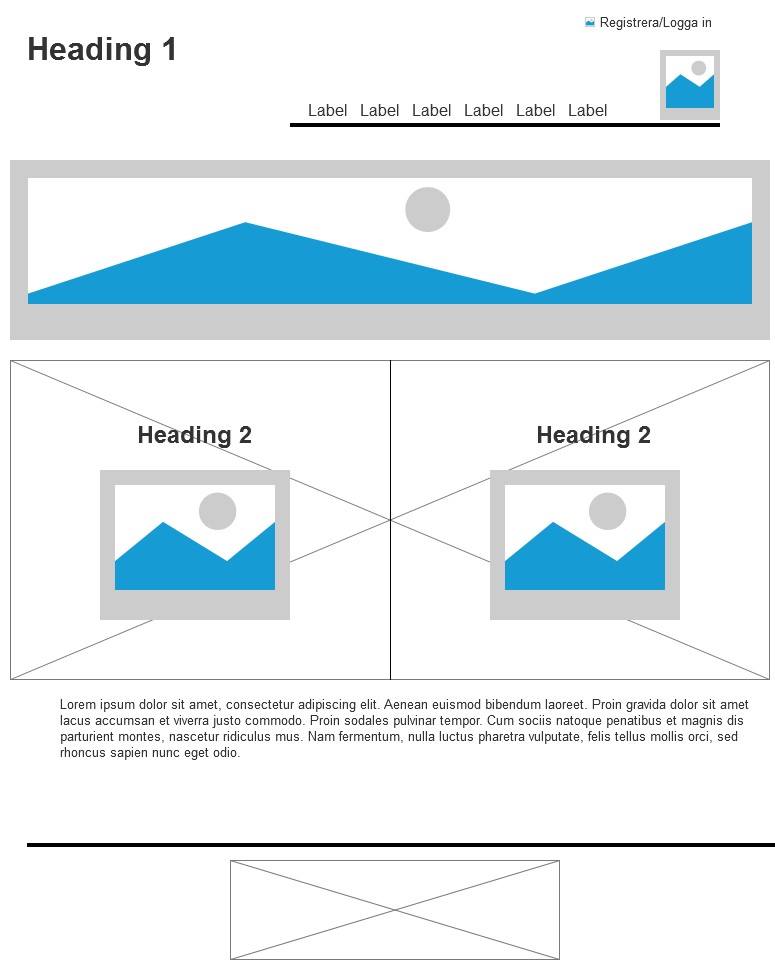
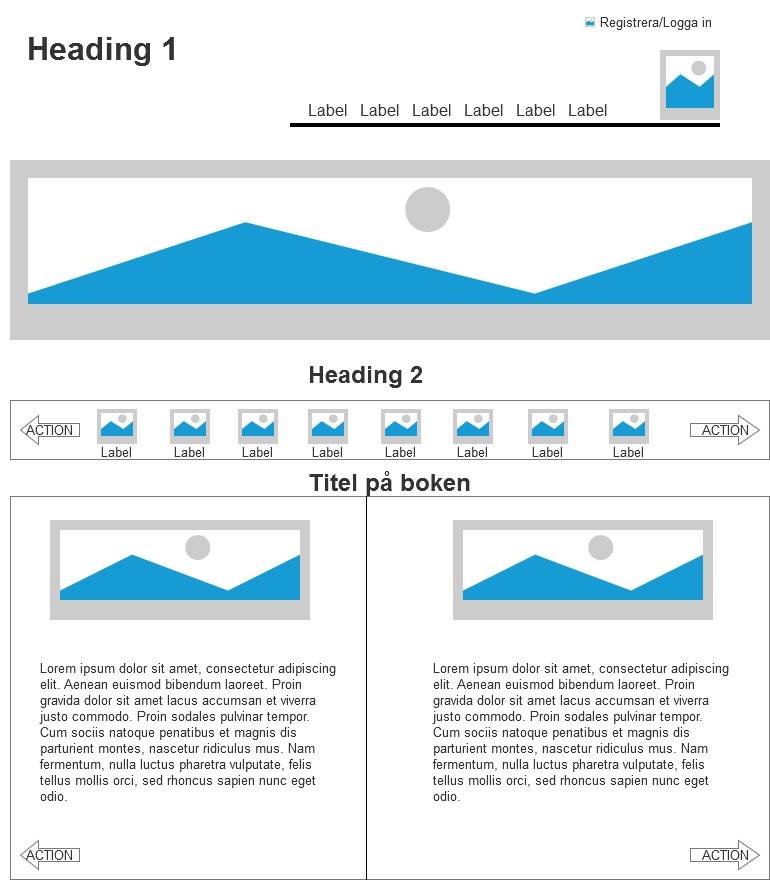
*Fig. 3.3 Skiss kontakta oss* *Fig. 3.4 Skiss indexsida*



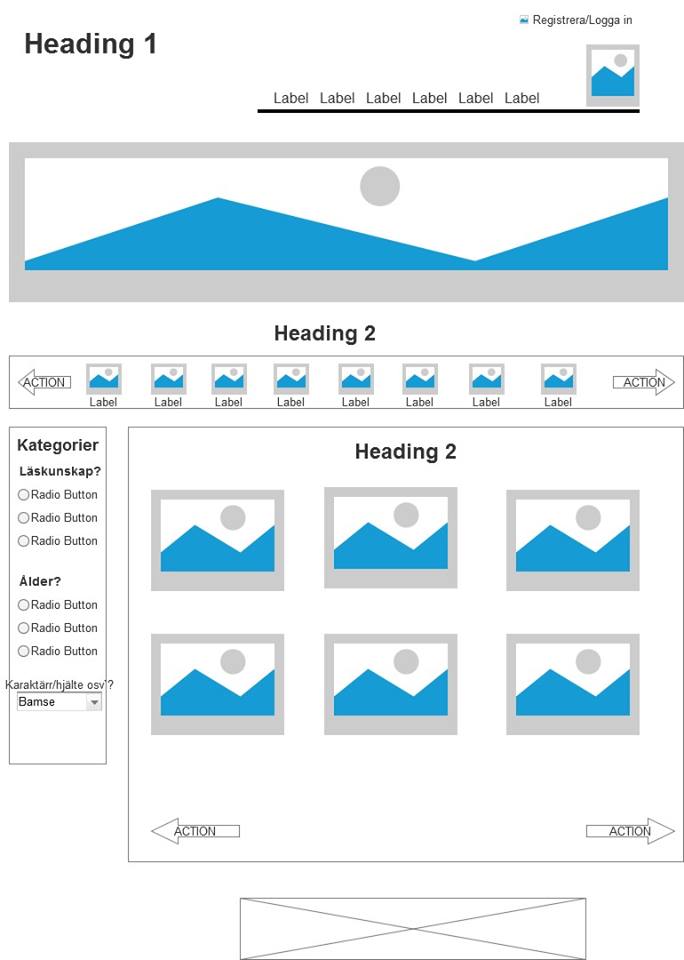
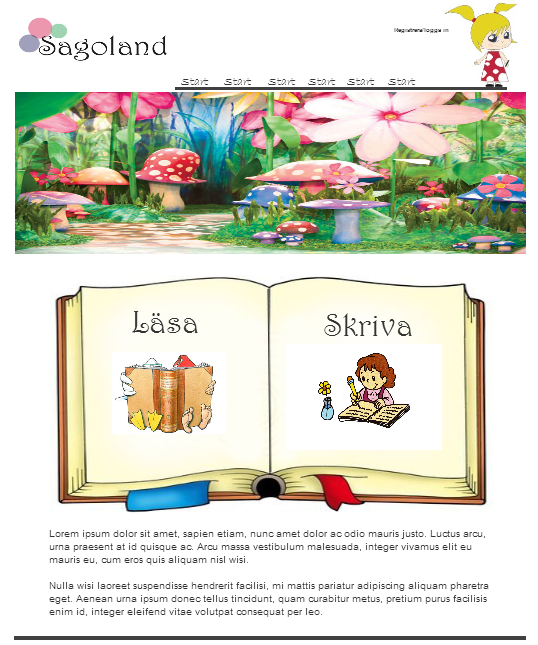
*Fig. 3.5 Skiss kategori* *Fig. 3.6 Skiss inloggning*

## 3.2 Wireframes

Syftet med att vi skapade våra wireframes var för att få en tydligare överblick för designens utseende. Målet var att arbeta så nära dessa wireframes som möjligt, allt för att underlätta arbetet för gruppen när det kom till vår design. Wireframes är inte vår slutgiltiga design utan enbart ett hjälpmedel i vår process mot den slutgiltiga designen (Se fig 3.7 – 3.9). Vi skapade en tydligare wireframe med hjälp av Justinmind där vi gav liv till vår wireframe med bilder och färg (Se fig 4.0).



*Fig 3.7 Sagolands indexsida* *Fig. 3.8 Sagolands läsfunktion*

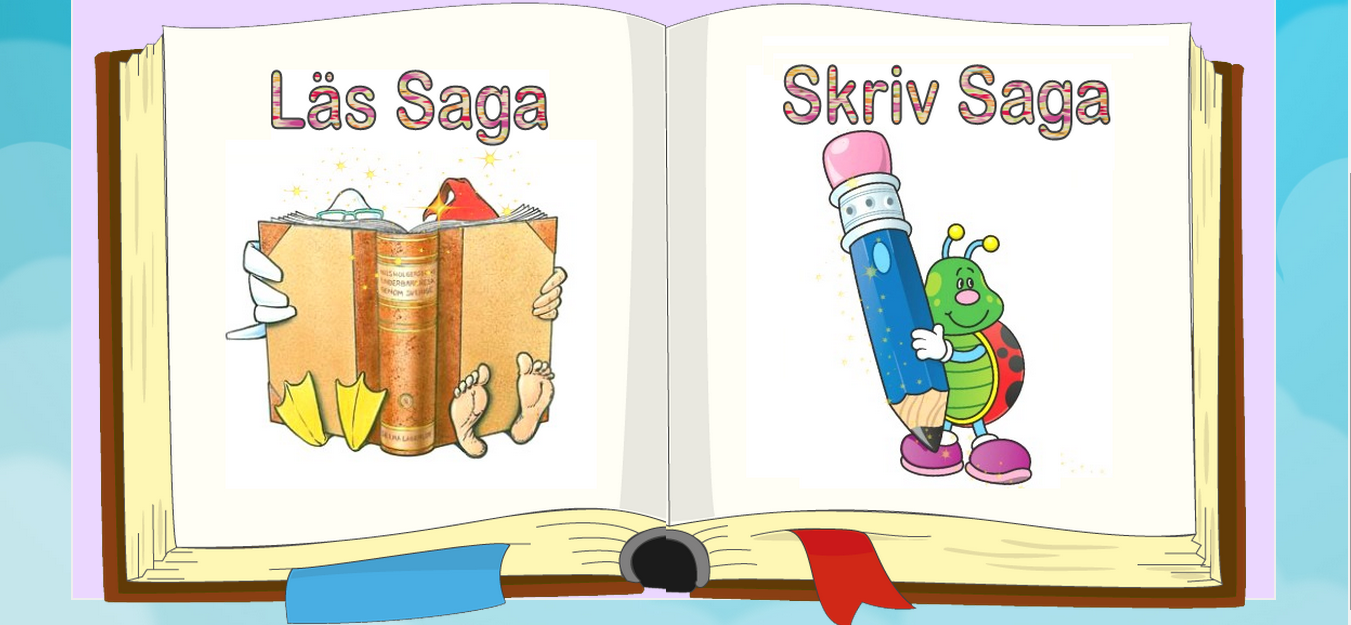


*Fig. 3.9* *Sagolands kategoriseringssida*  *Fig. 4.0 Sagolands indexsida*

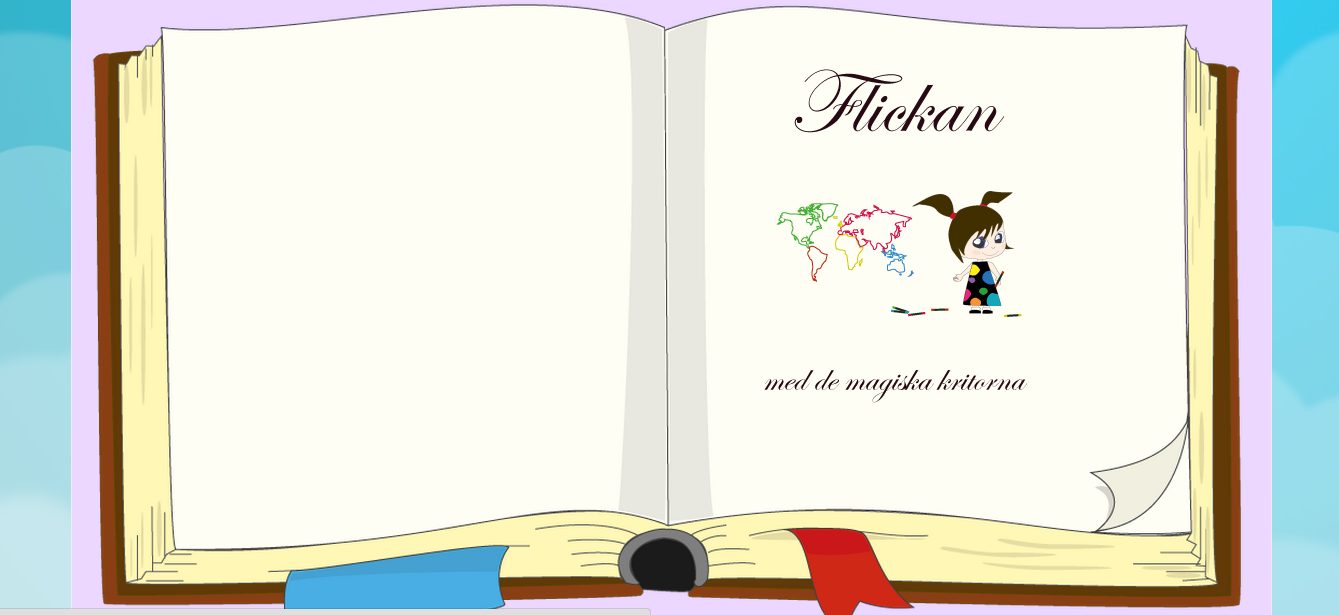
## 3.3 Slutgiltig design



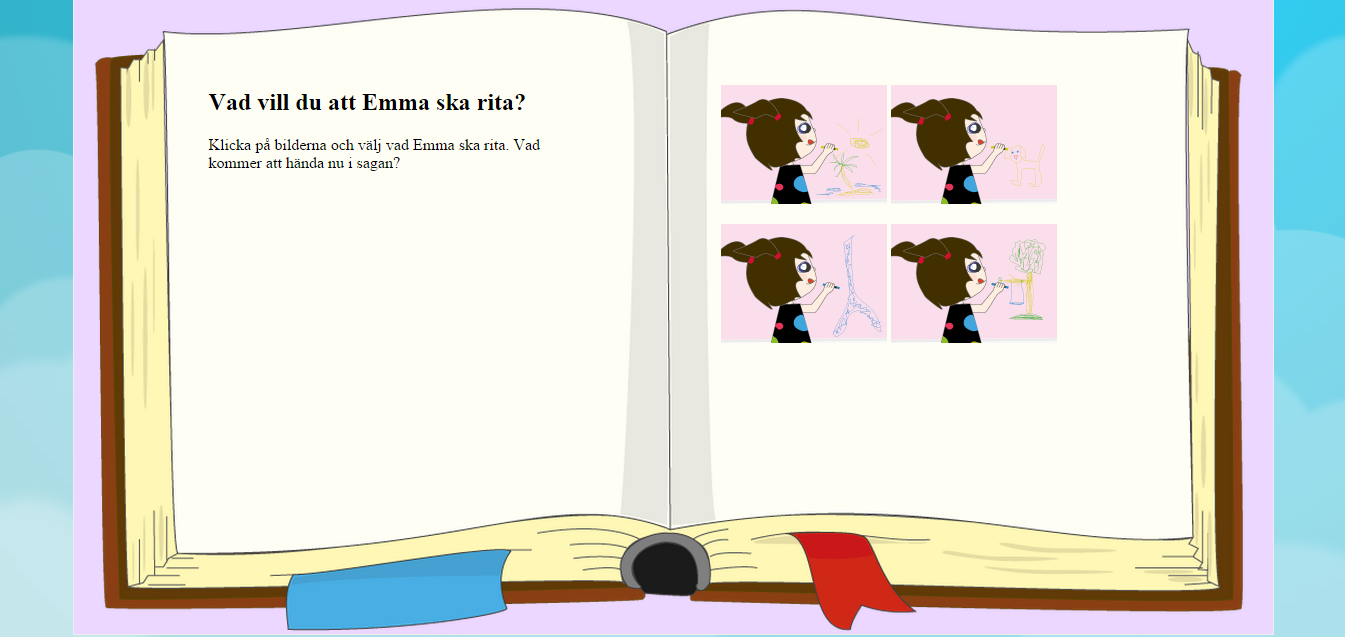
*Fig. 4.1 Sagolands header och navigations bar på indexsidan. Navigations baren finns på alla sidor i produkten.*

**

*Fig. 4.2 Sagolands enklare navigation på indexsidan. Med stor och tydlig text kan användaren klicka på helasidan för att ta sig vidare.*

**

*Fig. 4.3 Första sidan på en av våra sagor. När musen är på sidan går en flik upp som indikerar att användaren kan ”vända blad”.*

**

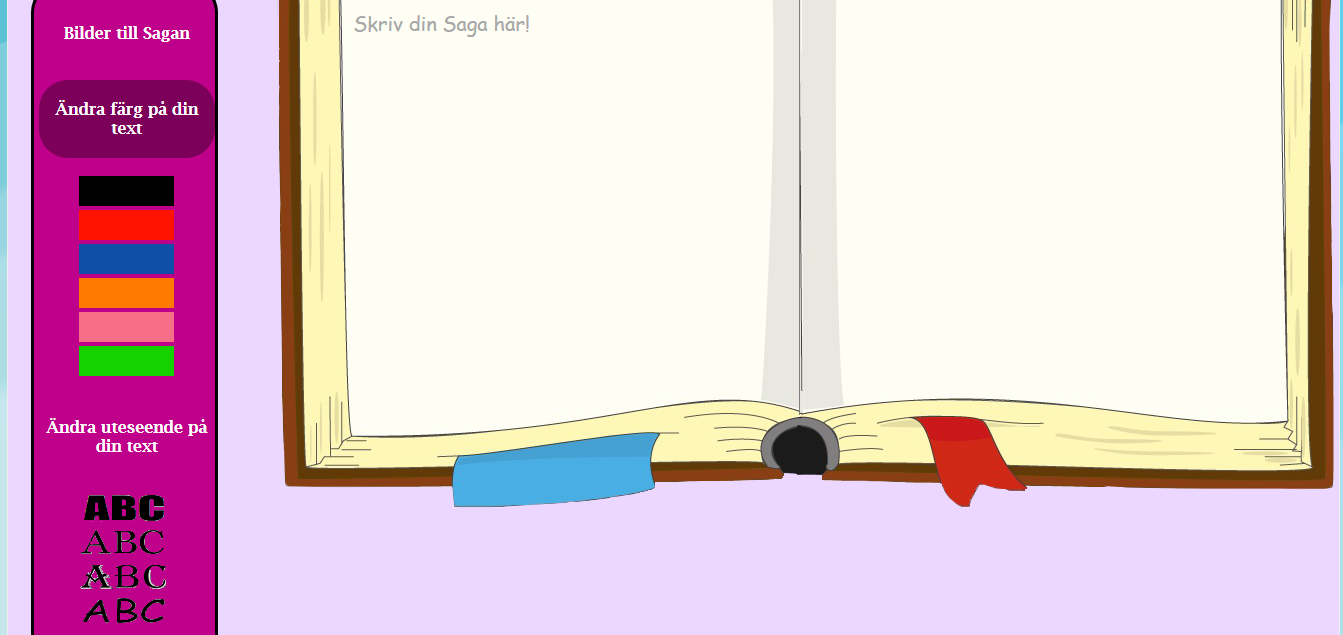
*Fig. 4.4 I sagan kan användaren själv välja vad ”Emma” ska rita.*

**

*Fig. 4.5 Inloggning/registrering funktionen. Detta måste användaren fylla i för att kunna använda sig av skrivfunktionen.*

**

*Fig. 4.6 Första sidan på skrivfunktionen. En stor tydlig meny till vänster som erbjuder olika funktioner som kan användas när användaren utformar sin saga. ”Vad heter din Saga?” Här kan användaren skriva sitt namn på sagan.*

**

*Fig. 4.7 Väldigt simpelt utformad meny som följer samma tema som den logiska navigationen med stora ytor att klicka på.*

**

*Fig. 4.8 I menyn kan användaren klicka på bilder för att få upp dessa bilder. Även här har vi valt att använda oss utav stora bilder och begränsa antalet för att hålla det simpelt.*