

Requirements

DAT255, Software engineering project
Chalmers, VT 2012

Patrik Ackerfors	861017-4990
Christoffer Kjernald	870820-5516
Erik Olesund	880917-2490
Markus Ulenius	880320-5635

Kraven är uppdelade på följande områden:

1. Skapa lista
2. Hantera lista
3. Skapa artikel
4. Hantera artikel
5. Delning

Requirement 1.1: Skapa en ny lista

Scenario:

Skapa en ny lista

Trigger:

Användaren väljer att skapa en ny lista.

Precondition:

Appen är startad

Standard case

Användaren fyller i namnet på listan (valfria tecken) och väljer sedan att skapa den. Systemet sparar den skapade listan.

Exceptions

1. Ingen titel anges
 - Knappen "Confirm" är inaktiv och det går inte att skapa listan
2. En titel får endast bestå av en rad
 - Knappen "Confirm" är inaktiv och det går inte att skapa listan
3. En titel får inte innehålla mer än 30 tecken
 - Knappen "Confirm" är inaktiv och det går inte att skapa listan

Result state

Listan skapas och läggs till i databasen

Requirement 2.1: Byta namn på lista

Scenario:

Byta namn på lista

Trigger:

Användaren väljer att byta namn på en lista

Precondition:

Listan existerar

Standard case

Användaren markerar den lista som denne önskar byta namn på, väljer funktionen byt namn, fyller i ett nytt namn och sparar ändringen. Systemet sparar ändringen och uppdaterar listans namn. Användaren förpassas till Main List activity.

Exceptions

1. Användaren raderar namnet så att listan saknar titel
 - Knappen "Confirm" är inaktiv och det går inte att göra ändringen
2. Användare försöker skriva in en textsträng som är längre än 30 tecken
 - Det är inte möjligt, det vill säga att tecken nummer 31 går inte att ange

Result state

All data i ursprungslistan finns kvar, men listan har nu ett nytt namn. Listan namn uppdateras i databasen.

Requirement 2.2: Radera en lista

Scenario:

Radera en lista

Trigger:

Användaren väljer att radera en lista

Precondition:

Listan existerar

Standard case

Användaren markerar den lista som denne önskar radera och håller fingret på den så att knappen "Delete" visas. Användaren trycker på delete och systemet raderar listan och användaren återgår till Main List activity. Om den raderade listan låg över någon annan, ska det inte finnas någon tom position mellan kvarvarande listor.

Exceptions

1. Det finns ingen lista att radera
 - Det går då inte att komma åt knappen "Delete" eftersom ingen lista kan markeras.

Result state

Listan raderas från databasen och finns inte längre kvar i Main List activity

Requirement 3.1: Lägg till ny artikel

Scenario:

Lägg till ny artikel

Trigger:

Användaren väljer att lägga till en artikel

Preconditions:

Den lista som användaren vill lägga till artikeln till existerar och användaren är inne i Main Item activityn för listan ifråga.

Standard case

Användaren fyller i titeln (obligatoriskt) på artikeln inuti den lista som artikeln ska tillhöra. Användaren beskriver artikeln med valfritt antal ord. Systemet adderar artikeln till den önskade listan.

Exceptions

1. Användaren anger ingen titel på artikeln
 - Knappen "Confirm" är inaktiv och artikeln går inte att skapa
2. Användaren försöker ange en titel som är längre än 20 tecken
 - Det är inte möjligt att skriva in det 21:a tecknet

Result state

Artikeln skapas och läggs till i databasen. Den finns synlig längst ner bland artiklarna i listan

Requirement 4.1: Byta namn på artikel

Scenario:

Byta namn på artikel

Trigger:

Användaren väljer att byta namn på en artikel

Precondition:

Artikeln existerar

Standard case

Användaren markerar den artikel som denne önskar byta namn på till dess att knapparna "Delete" och "Edit" dyker upp. Användaren trycker på "Edit", skriver in det nya namnet och trycker på "Confirm". Systemet sparar ändringen och uppdaterar artikelns namn.

Exceptions

1. Användaren raderar namnet så att artikeln saknar titel
 - Knappen "Confirm" är inaktiv och det går inte att göra ändringen
2. Användaren försöker ange en ny titel som innehåller fler än 20 tecken
 - Det är inte möjligt att skriva in det 21:a tecknet

Result state

Artikelns namn uppdateras i databasen och den nya titeln syns i "Main Item activity" för listan ifråga.

Requirement 4.2: Lägga till extra information om en artikel / Byta beskrivning**Scenario:**

Lägga till extra info om en artikel / byta beskrivning

Trigger:

Användaren väljer att lägga till (ytterligare) information om en artikel

Precondition:

Listan existerar och artikeln existerar.

Standard case

Användaren markerar den artikel som denne önskar ändra beskrivningen på. Användaren fyller i önskad beskrivning i fältet "Description" och trycker på "Confirm". Systemet sparar ändringen och uppdaterar artikelinformationen. Användaren förpassas till Main Item activity.

Exceptions

Det existerar inga exceptions.

Result state

Databasen uppdateras med information om artikeln ifråga

Requirement 4.3: Boka artikel**Scenario:**

Boka artikel

Trigger:

Användaren markerar en artikel på listan som bokad.

Precondition:

Artikeln existerar och har inte bokats tidigare.

Standard case

Användaren markerar den artikel som den vill boka. Systemet sparar bokningen och knappen ändrar utseende.

Exceptions

Det finns inga exceptions.

Result state

Artikeln status-ruta anger "Bokad" och databasen uppdateras med informationen

Requirement 4.4: Avboka artikel

Scenario:

Avboka artikel

Trigger:

Användaren markerar en artikel på listan som avbokad

Precondition:

Artikeln existerar och har statusen "Bokad" i status-rutan.

Standard case

Användaren markerar den artikel som den vill avboka och gör detta. Systemet sparar ändringen att artikeln är avbokad och har statusen "Boka" i status-rutan.

Exceptions

1. Artikeln har statusen "Boka" i status-rutan
- Knappen "Bokad" finns inte utan är ersatt av statusen "Boka"

Result state

Artikeln status-ruta anger "Boka" och databasen uppdateras med informationen

Requirement 4.5: Radera en artikel

Scenario:

Radera en artikel

Trigger:

Användaren väljer att radera en artikel

Precondition:

Artikeln existerar

Standard case

Användaren markerar den artikel som den vill radera och gör detta. Systemet sparar ändringen av att artikeln är raderad från listan. Övriga artiklar placerar om sig så att de ligga efter varandra utan en tom position.

Exceptions

Det finns inga exceptions

Result state

Artikeln raderas från listan och databasen uppdateras

Requirement 5.1: Dela en lista

Scenario:

Dela en lista

Trigger:

Användaren väljer att dela en lista.

Precondition:

Listan existerar

Personen/(-rna) (mottagaren/-arna) som användaren vill dela listan med måste ha en fungerande e-mailadress

Telefonen är uppkopplad mot internet.

Standard case

Användaren markerar den lista som den vill dela. Användaren skriver in mottagarens/(-arnas) e-mailadress separerat med "," i fältet "Email" och bekräftar att den vill dela listan.

Användaren kan välja att ange både "subject" och "message"

Exceptions

1. Den angivna e-mailadressen är ogiltig, dvs innehåller inte ett @ och en "." (punkt).

- Systemet talar om för användaren att listan inte delats och uppmanar att skriva in adressen på nytt.

2. Den angivna e-mailadressen innehåller de obligatoriska tecknen men är dock inte en fungerande eller korrekt adress.

- Systemet skickar listan, men bekräftar dock inte om den har blivit korrekt mottagen eller ej

Result state

Mottagaren/(-arna) får ett e-mail med en listans titel samt samtliga items som tillhör listan.

Future requirements

Requirement 5.2: Listigt-användare ska kunna ta emot en lista i sin applikation

Scenario:

En listigt-användare tar emot listan i sin applikation

Trigger:

Användaren trycker på en länk i e-målet

Precondition:

En skapare av en lista har delat listan till en annan användare av applikationen.

Standard case

Användaren trycker på en länk i e-målet och förflyttas då till sin app där listan visas i enlighet

med interfacet för appen.

Exceptions

Det finns inga exceptions.

Result state

Listan har delats och båda skaparen och mottagaren av listan kan nu se listan med dess innehåll i appen.

Requirement 5.3: Interaktion mellan användare av samma lista

Scenario:

Listan uppdateras för samtliga

Trigger:

En användare bokar, avbokat, raderar eller adderar en artikel i en lista

Precondition:

Listan existerar

Listan har delats mellan två eller flera användare

Standard case

Då en användare bokar, avbokat, raderar eller adderar en artikel i en lista ska det synkroniseras för samtliga användare som listan har delats med

Exceptions

Det går bara att avboka de artiklar som man som användare själv har bokat

Det är bara skaparen av en lista som kan radera eller addera artiklar till en lista

Result state

Databasen uppdateras och förändringen synliggörs för samtliga användare

Requirement 5.4: Behörighetsnivån för en lista ska kunna regleras

Scenario:

Skaparen av listan väljer att reglera sin egen åtkomst till listans artiklars status, det vill säga om de är boka eller inte och av vem. Listan kan då vara av två varianter där den först innebär att skaparen har fri åtkomst till listan och dess status och den andra innebär att skaparen inte har åtkomst till listans status efter att den delats.

Trigger:

Användaren reglerar sin egen behörighet när listan skapas. Det görs genom att i activityn ListEditor bestämma vilken typ av lista det är.

Precondition:

Appen är startad.

Standard case

Användaren skapar en lista och väljer samtidigt behörighetsnivån för listan. Detta ligger till grund för all framtida delning av listan till andra användare.

Exceptions

Det finns inga exceptions.

Result state

Behörighetsnivån av listan är bestämd.

Requirement 5.4: Skaparen av en lista ska inte kunna se förändringar i listan, så som när en artikel bokas eller avbokas

Scenario:

Ett födelsedagsbarn har skapat en lista och delat den med sina gäster. En gäst väljer att boka/köpa en av artiklarna/presenterna i listan. Födelsedagsbarnet ska då inte se om artikeln har bokats eller ej

Trigger:

Användaren bokar en artikel

Precondition:

Appen är startad

Listan är skapad och delad med användaren som utför bokningen och ytterligare minst en person

Listan innehåller minst en artikel

Skaparen av har valt att listans behörighetsnivå ska vara reglerad så att han/hon inte ser förändringar i listan

Standard case

En användare (ej skaparen) bokar en artikel. För samtliga användare av listan syns att artikeln har blivit bokad, men för skaparen av listan händer ingenting

Exceptions

Det finns inga exceptions.

Result state

Databasen uppdateras och för samtliga användare (utom skaparen) syns att artikeln blivit bokad, men för skaparen av listan sker ingenting