

# *Kravspekifikation*

---

## *Autonom overvågningsdrone*

Udarbejdet af	Rasmus Lydixsen Anders Opstrup Kevin Grooters
Vejleder	Torben Gregersen
Projektnummer	14123
Antal sider	15

---

Rasmus Lydixsen  
11647

---

Anders Opstrup  
11726

---

Kevin Grooters  
11655

# Indholdsfortegnelse

---

<b>Kapitel 1</b>	<b>Intro</b>	<b>1</b>
1.1	Revisionshistorik . . . . .	1
1.2	Ordforklaring . . . . .	1
1.3	Indledning . . . . .	1
1.4	Systembeskrivelse . . . . .	2
1.5	Systemoversigt . . . . .	2
1.6	Brugsscenarie . . . . .	3
1.7	Prioritering . . . . .	3
<b>Kapitel 2</b>	<b>Funktionelle krav</b>	<b>4</b>
2.1	Aktør diagram . . . . .	4
2.2	Aktørbeskrivelser . . . . .	4
2.3	Use case diagram . . . . .	5
2.4	Udviklingsforløb . . . . .	6
2.5	Use case beskrivelse . . . . .	7
<b>Kapitel 3</b>	<b>Ikke-funktionelle krav</b>	<b>12</b>
<b>Kapitel 4</b>	<b>Grænseflader</b>	<b>13</b>
4.1	Brugergrænseflade . . . . .	13
4.2	Webapplikation . . . . .	13

## 1.1 Revisionshistorik

Rev. Nr	Dato	Initialer	Ændring
1.0	25-08-14	KG,RL,AO	Oprettet kravspecifikations dokument.
1.1	03-09-14	KG,RL,AO	Use cases opdateret.
1.2	09-09-14	KG,RL	Opdateret kravspecifikation.

*Tabel 1.1:* Revisionshistorik

## 1.2 Ordforklaring

Forkortelse	Betydning	Forklaring
UC	Use case	Beskrivelse af hvordan system bruges og fungerer
Fejlmode	Der opstår uventede fejl	Fejlmode er en tilstand dronen går i. Se protokol i bilag for mere info.

*Tabel 1.2:* Ordforklaring

## 1.3 Indledning

Dette dokument indeholder kravspecifikation tilhørende projektet: *Autonom overvågnings-drone*. Krav og egenskaber for systemet beskrives i dokumentet.

## 1.4 Systembeskrivelse

På Ingeniørhøjskolen Aarhus Universitet forefindes en AeroQuad Cyclone ARF Quadcopter. Målet med projektet er, at omdanne quadcopteren til en autonom overvågnings-drone.

Bruger skal kunne tilgå dronen via en webapplikation, som skal fungere som en grafisk brugerflade mellem bruger og drone. Via webapplikationen skal bruger blandt andet kunne lave opsætning til nye flyvninger, se billeder og flyverute fra tidligere flyvninger.

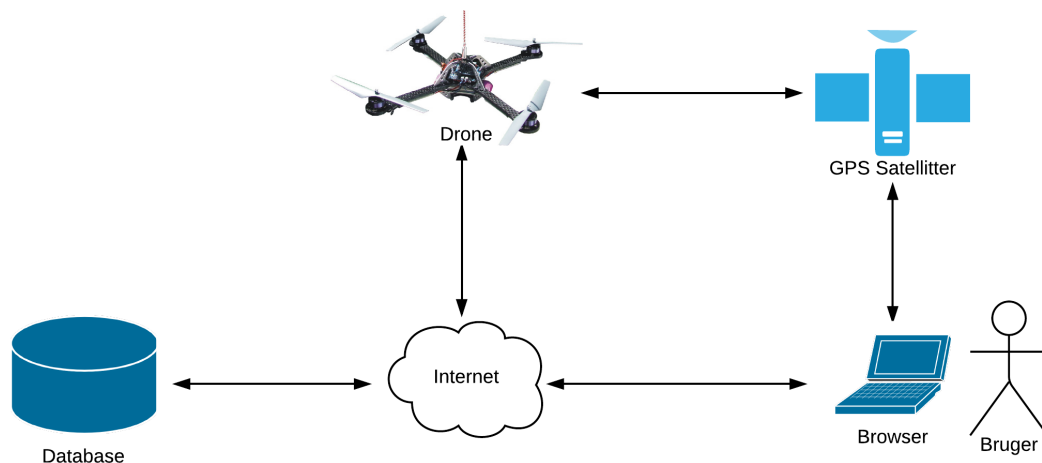
Ved opsætning til ny flyvning vælger bruger en række GPS positioner som dronen skal flyve til. Desuden skal bruger vælge hvorvidt der skal tages billeder ved de valgte GPS positioner.

Til enhver tid, skal al kommunikation mellem webapplikation og drone foregå via mobil netværk - hovedsageligt 3G. Da dronen skal flyve autonomt, er det vigtigt den kan orientere sig på egen hånd. Derfor er den udstyret med GPS, afstands sensorer og kompas.

## 1.5 Systemoversigt

Nederst til højre på systemskitsen ses et device med internet adgang. Dette devices bruges af bruger til at tilgå webapplikation, hvor opsætning af ny flyvning ordnes. Når en bruger har lavet indstillinger til ny flyvning, overføres opsætningen via internettet til dronen.

Via kommunikation med GPS satellitter finder dronen frem til egen position og hvilken retning den skal flyve i. Under flyvningen kan tage dronen billeder, som via nettet overføres til en database der er tilknyttet webapplikationen.



*Figur 1.1:* Systemskitse

## 1.6 Brugsscenarie

Et typisk brugsscenarie for den autonome overvågningsdrone vil være overvågning af et privat område eller virksomhed. Dronen kan nemt og omkostningsfrit overvåge et større område, hvilket betyder at brug af dronen vil spare menneskelige ressourcer.

Dronen kræver to ting for at fungere. Dels skal den regelmæssigt have udskiftet eller opladet sit batteri, og desuden skal den have besked om hvor den skal overvåge.

Dronen opsættes via en webapplikation. Fra webapplikationen kan bruger indtaste hvor højt dronen skal flyve, hvilken højde der skal tages overvågningsbilleder fra og hvor der skal tages overvågningsbilleder.

## 1.7 Prioritering

Område	Prioritering	Kommentar
Sikkerhed	5	Sikkerheden prioriteres højst, idet dronens propeller er farlige og kan skade mennesker og dyr.
Pålidelighed	4	Hele systemet skal være meget pålidelig, da dronen under flyvning aldrig må fejle. Da fejl eller svigt kan udsætte mennesker og dyr for fare.
Pris	2	Prisen er mindre vigtig, da der er stor interesse for udviklingen af denne type overvågning.
Brugervenlighed	3	Systemet skal ikke kunne betjenes af alle, derfor er brugervenligheden ikke den vigtigste prioritering.

**Tabel 1.3:** Prioriteringsliste

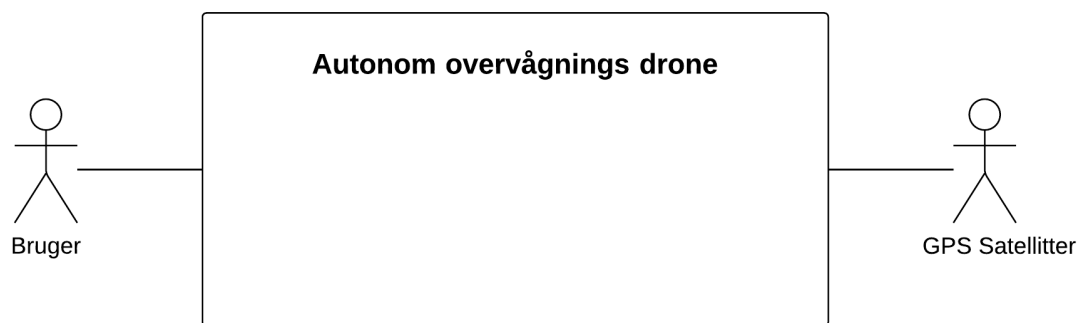
# Funktionelle krav 2

---

De funktionelle krav beskrives via brugsscenarier, også kaldet use cases. Indledningsvis beskrives systemets aktører, og senere i afsnittet beskrives hvordan systemet fungerer ud fra interaktion mellem aktører og system.

## 2.1 Aktør diagram

Nedenstående figur viser hvilke aktører der interagerer med systemet.



*Figur 2.1:* Aktør diagram

## 2.2 Aktørbeskrivelser

Aktørbeskrivelsen skitserer systemets aktører samt hvilken rolle de spiller for systemet.

Navn	Bruger.
Type	Primær.
Beskrivelse	Bruger er den eneste person der interagerer med systemet. Via webapplikation indstiller bruger flyveopsætning for nye flyvninger. samt undersøge billeder og flyveruter fra tidligere flyvninger.

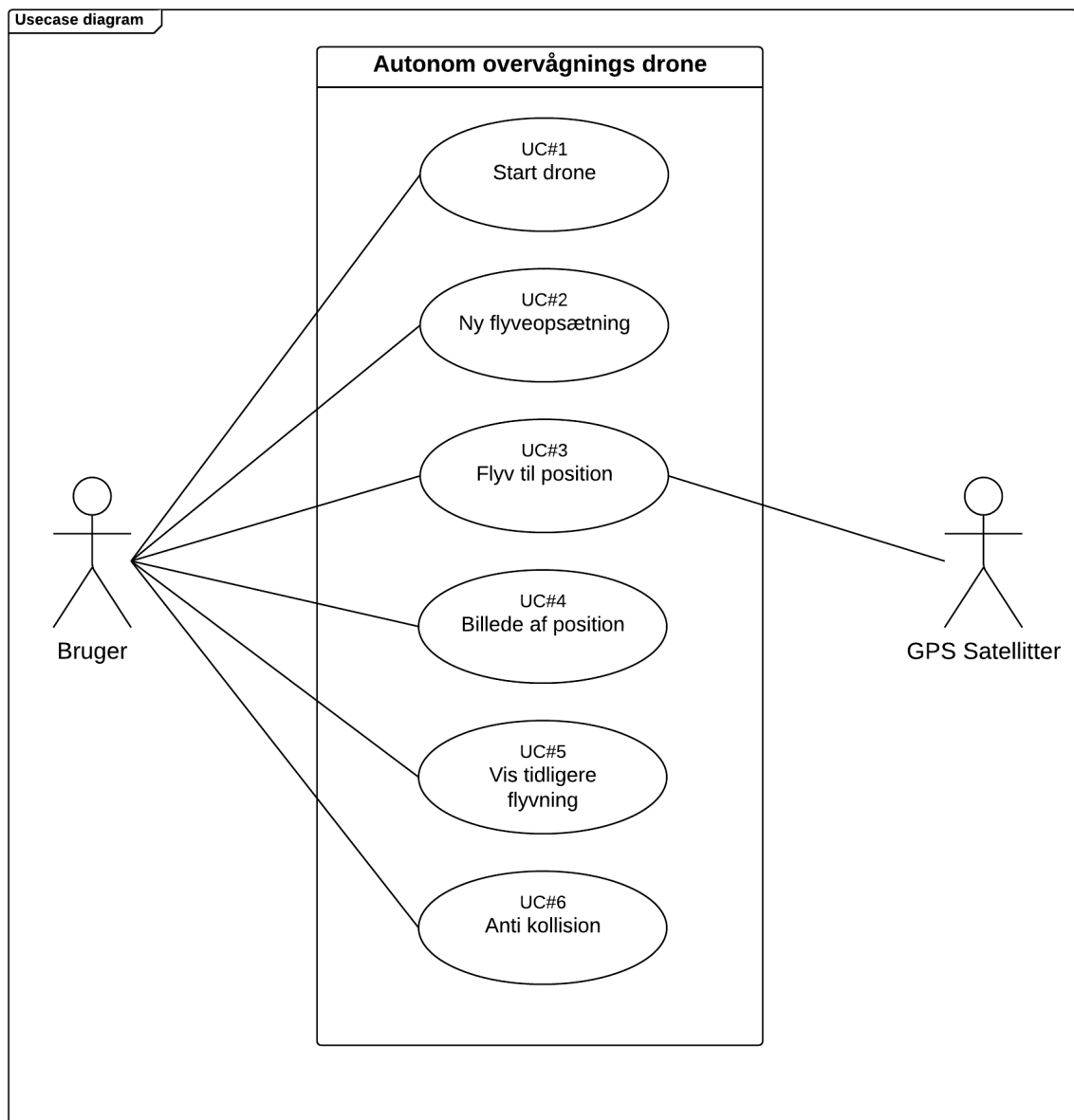
*Tabel 2.1:* Aktørbeskrivelse, Bruger

Navn	GPS satellitter.
Type	Sekundær.
Beskrivelse	GPS satellitterne bruges når dronen lokaliserer sin position.

*Tabel 2.2:* Aktørbeskrivelse, GPS satellitter

## 2.3 Use case diagram

På figur 2.2 vises de identificerede use cases.



*Figur 2.2:* Use case diagram

## 2.4 Udviklingsforløb

For at gøre udviklingsforløbet mere overskueligt gøres brug af iterationer. De iterationer der ligger tidligst i udviklingsforløbet er mest essentielle for systemet, mens de iterationer der ligger senere i forløbet prioriteres knap så højt. Udviklingsforløbet i dette projekt er planlagt til at forløbe via 4 iterationer. Nedenfor beskrives hvad der skal laves i de forskellige iterationer.

### Iteration 1:

I denne iteration tages hånd om systemet mest grundlæggende funktionalitet. Batteri, ESC'er, motorer og ultralyds sensorer tilsluttes dronen. Desuden gøres dronen i stand til at kunne oprette forbindelse til webapplikation via 3G-shield'et. Når iteration 1 er færdig skal UC#1 kunne gennemføres.

### Iteration 2:

I iteration 2 er hovedformålet at få styr på kommunikationen mellem webapplikation og drone. Det skal være muligt for bruger både at oprette og sende flyveopsætning til dronen. Ydermere skal drone kunne finde egen GPS position, flyvehøjde og orientering. Ud fra viden om egen position, flyvehøjde og orientering skal dronen kunne flyve til de lokationer bruger ønsker. Når iteration 2 er færdig skal UC#2 og UC#3 kunne gennemføres.

### Iteration 3:

I iteration 3 er det primære fokus billeder. Der skal monteres kamera på dronen, så den kan tage billeder ved de lokationer som bruger har defineret. Alle billeder taget under flyvning sendes via mobilnet fra dronen til webapplikation og gøres tilgængelige for bruger. Når iteration 3 er færdig skal UC#4 og UC#5 kunne gennemføres.

### Iteration 4:

I iteration 4 ønskes det at udvikle anti kollision til dronen. Inden tilføjelsen af anti kollision kan dronen udelukkende flyve i lukkede områder uden forhindringer. Tilføjelse af anti kollision vil muliggøre flyvning i normale områder med forhindringer. Når iteration 4 er færdig skal UC#6 kunne gennemføres



## 2.5 Use case beskrivelse

I afsnittet vises fully dressed use case beskrivelser af de use cases vist i på figur 2.3.

### UC1 - Start drone

Mål	Drone er tændt og har forbindelse til webapplikation.
Initiering	Bruger.
Aktører og interesserede	Bruger (primær) <ul style="list-style-type: none"><li>• Bruger tænder drone.</li></ul>
Startbetingelser	Ingen.
Slutbetingelser	Drone kan modtage flyveopsætning fra webapplikation.
Hovedforløb	<ol style="list-style-type: none"><li>1. Bruger tilslutter batteri og dronen tændes.</li><li>2. Main controller initialiseres.</li><li>3. GPS initialiseres og dronens nuværende GPS position opdateres.</li><li>4. 3G initialiseres og dronen opretter forbindelse til 3G-netværket.</li></ol> <p>I: Forbindelse kan ikke oprettes.</p> <ol style="list-style-type: none"><li>5. Dronens nuværende GPS position sendes til webapplikation.</li></ol>
Undtagelser	<p>I: Forbindelse kan ikke oprettes.</p> <p>a) Systemet indikerer fejl og derfor genstarter bruger systemet. Derefter genstartes UC#1</p>

**Tabel 2.3:** Use Case 1

## UC2 - Ny flyveopsætning

Mål	Ny flyveopsætning er oprettet og sendt til drone.
Initiering	Bruger.
Aktører og interesserede	Bruger (primær) <ul style="list-style-type: none"> <li>• Bruger ønsker at få adgang til webapplikation.</li> <li>• Fra webapplikation indstiller bruger flyveopsætning.</li> <li>• Bruger sender flyveopsætning til drone.</li> </ul>
Startbetingelser	Bruger er oprettet i systemet og UC#1 er succesfuld gennemført
Slutbetingelser	Ny flyveopsætning er sendt til drone.
Hovedforløb	<ol style="list-style-type: none"> <li>1. Bruger logger på webapplikation. <ol style="list-style-type: none"> <li>I: Fejl i login.</li> </ol> </li> <li>2. Fra forsiden navigerer bruger til flyveopsætning.</li> <li>3. Bruger laver en ny flyveopsætning og vælger: <ul style="list-style-type: none"> <li>- GPS lokationer der skal flyves til.</li> <li>- Om der skal tages billeder ved de valgte lokationer.</li> <li>- Højde billeder skal tages fra.</li> <li>- Generel flyvehøjde.</li> </ul> <ol style="list-style-type: none"> <li>I: Der laves ikke ny flyveopsætning.</li> </ol> </li> <li>4. Bruger gemmer flyveopsætning.</li> </ol>
Undtagelser	<ol style="list-style-type: none"> <li>I: Fejl i login. <ol style="list-style-type: none"> <li>a) Bruger føres tilbage til login.</li> </ol> </li> <li>II: Der laves ikke ny flyveopsætning. <ol style="list-style-type: none"> <li>a) Gemt flyveopsætning benyttes.</li> </ol> </li> </ol>

**Tabel 2.4:** Use Case 2

## UC3 - Flyv til position

Mål	Drone flyver til ønsket position.
Initiering	UC#2 eller UC#4.
Aktører og interesserede	Bruger (primær) <ul style="list-style-type: none"> <li>• Bruger ønsker at drone flyver som angivet i flyveopsætning.</li> </ul> GPS satellitter (sekundær) <ul style="list-style-type: none"> <li>• Dronen opdaterer egen GPS position vha. GPS satellitterne.</li> </ul>
Startbetingelser	UC#1 og UC#2 er succesfuld gennemført.
Slutbetingelser	Position er nået.
Hovedforløb	<ol style="list-style-type: none"> <li>1. Drone henter flyveopsætning fra server.               <ul style="list-style-type: none"> <li>I: Dronen henter ikke flyveopsætning, da flyvning med en anden flyveopsætning er aktiv.</li> </ul> </li> <li>2. Nuværende position opdateres.               <ul style="list-style-type: none"> <li>I: Ugyldig GPS koordinat.</li> </ul> </li> <li>3. Flyvehøjde tilpasses.               <ul style="list-style-type: none"> <li>I: Ugyldig flyvehøjde.</li> </ul> </li> <li>4. Flyveorientering tilpasses.</li> <li>5. Drone flyver mod ønsket position i ønsket flyvehøjde.</li> <li>6. Ønsket position er nået.               <ul style="list-style-type: none"> <li>I: Ønsket position er ikke nået.</li> </ul> </li> </ol>
Undtagelser	I: Ugyldig GPS koordinat. <ol style="list-style-type: none"> <li>a) Drone går i fejlmode #1.</li> </ol> II: Ugyldig flyvehøjde. <ol style="list-style-type: none"> <li>a) Drone går i fejlmode #2.</li> </ol> III: Ønsket position er ikke nået. <ol style="list-style-type: none"> <li>a) UC #3 genstartes.</li> </ol>

**Tabel 2.5:** Use Case 3

## UC4 - Billede af position

Mål	Drone tager et billede af nuværende position som sendes til webapplikation. Fra webapplikation kan bruger se og acceptere billedet.
Initiering	UC#3.
Aktører og interesserede	Bruger (primær) <ul style="list-style-type: none"> <li>• Kan inspicere og acceptere billede.</li> </ul>
Startbetingelser	UC#1, UC#2 og UC#3 er succesfuld gennemført.
Slutbetingelser	<ul style="list-style-type: none"> <li>• Bruger kan tilgå billedet via webapplikation.</li> <li>• Drone flyver til næste GPS-position eller udgangsposition.</li> </ul>
Hovedforløb	<ol style="list-style-type: none"> <li>1. Drone tager et billede af nuværende position.</li> <li>2. Billedet sendes til webapplikation.</li> <li>3. Bruger giver accept af billedet via webapplikation. <ol style="list-style-type: none"> <li>I: Bruger beder om et nyt billede.</li> <li>II: Bruger svarer ikke inden for tidsgrænsen.</li> </ol> </li> <li>4. Drone sendes information om næste lokation.</li> </ol>
Undtagelser	<ol style="list-style-type: none"> <li>I: Bruger beder om et nyt billede. <ol style="list-style-type: none"> <li>a) Drone instrueres til at ændre højde, orientering eller position. Derefter genstartes UC#4.</li> </ol> </li> <li>II: Bruger svarer ikke inden for tidsgrænsen. <ol style="list-style-type: none"> <li>a) Drone får automatisk tildelt accept.</li> </ol> </li> </ol>

**Tabel 2.6:** Use Case 4

## UC5 - Vis tidligere flyvning

Mål	Bruger tilgår webapplikation hvor tidligere flyveruter og tilhørende billeder forefindes.
Initiering	Bruger.
Aktører og interesserede	Bruger (primær) <ul style="list-style-type: none"> <li>• Ønsker at få adgang til webapplikation.</li> <li>• Fra webapplikation kan bruger undersøge flyveruter og billeder fra tidligere flyvninger.</li> </ul>
Startbetingelser	Bruger er oprettet i systemet.
Slutbetingelser	Ingen.
Hovedforløb	<ol style="list-style-type: none"> <li>1. Bruger logger på webapplikation. <ul style="list-style-type: none"> <li>I: Fejl i login.</li> </ul> </li> <li>2. Fra forsiden navigerer bruger til historik.</li> <li>3. Bruger vælger en specifik tidligere flyvning.</li> </ol>
Undtagelser	I: Fejl i login. <ol style="list-style-type: none"> <li>a) Bruger føres tilbage til login skærm.</li> </ol>

**Tabel 2.7:** Use Case 5

## UC6 - Anti kollision

Mål	Drone kan undvige forhindringer vha. anti kollision
Initiering	Ingen - altid aktiv.
Aktører og interesserede	Bruger (primær) <ul style="list-style-type: none"> <li>• Ønsker flyvning uden kollision.</li> </ul>
Startbetingelser	UC#3 er igangværende.
Slutbetingelser	Drone har undveget en kollision.
Hovedforløb	<ol style="list-style-type: none"> <li>1. Anti kollision detekterer en forhindring.</li> <li>2. Undvigningsmanøvre udføres. <ul style="list-style-type: none"> <li>I: Forhindring kan ikke undviges.</li> </ul> </li> </ol>
Undtagelser	I: Forhindring kan ikke undviges. <ol style="list-style-type: none"> <li>a) Drone går i fejlmode #3.</li> </ol>

**Tabel 2.8:** Use Case 6

# Ikke-funktionelle krav 3

---

De ikke-funktionelle krav indeholder specifikke krav som timings, afstande og lydniveauer. Ikke-funktionelle inddeles i følgende 3 grupper: Generelle krav, krav til webapplikation og krav til drone.

## 1. Generelle krav

- 1.1. Kommunikation mellem drone og webapplikation skal foregå trådløst.
- 1.2. Trådløs kommunikation benytter 3G protokol eller ældre.
- 1.3. Højdemåler skal måle højde  $\pm 10$  cm.

## 2. Krav til webapplikation

- 2.1. Webapplikation skal kunne tilgås via både computere og telefoner.
- 2.2. Indholder database med billeder og flyveruter fra tidligere.
- 2.3. Indholder database med brugere.

## 3. Krav til dronen

- 3.1. Skal forsynes fra batteri.
- 3.2. Batterilevetiden skal minimum være 15 minutter.
- 3.3. Flyvehastigheden skal minimum være  $2\frac{m}{s}$ .
- 3.4. Flyvehøjde kan reguleres i følgende 3 intervaller: 1-1.5m, 1.5-2m og 2-2.5m.
- 3.5. Højde der tages billeder fra kan reguleres mellem 1 og 2,5 meter.

## 4. Krav til opsamling af data

- 4.1. Tiden mellem et billede tages og til det er tilgængeligt på webapplikation skal maksimalt være 5 sekunder.
- 4.2. Gyldig højdemåling ligger i intervallet 0,5 til 4,5 meter.
- 4.3. GPS skal angive koordinat indenfor  $\pm 2,5$  meter.

# Grænseflader 4

---

Dette afsnit bruges til at beskrive systemets grænseflader og forklare hvordan bruger kan interagere med systemet.

## 4.1 Brugergrænseflade

Brugergrænsefladen består af en webapplikation, hvor bruger kan opsætte nye flyveruter, gemme flyveruter og overvåge drone status. Webapplikationen indeholder en database, som muliggøre arkiv funktioner for bruger.

## 4.2 Webapplikation

Webapplikationen er grænseflade mellem bruger og den resterende del af systemet.

På figur 4.1 ses login vinduet, som bruger bliver præsenteret for når der ønskes at interagere med systemet. Her vil bruger kunne logge ind og opsætte flyveruter, se gamle ruter og billeder fra tidligere flyvninger. Der accepteres kun tre fejl indtastninger når koden til systemet indtastes. Hvis disse tre forsøg overskrides bliver brugeren låst og der skal kontaktes en administrator for at låse den pågældende bruger op igen.



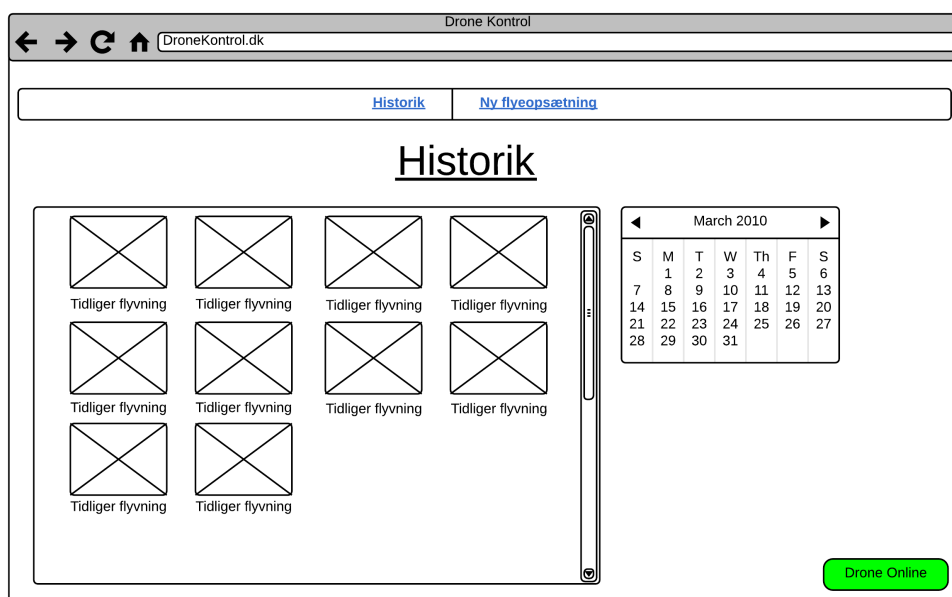
*Figur 4.1:* Login vindue

Efter login præsenteres bruger for webapplikations forside, se figur 4.2. Fra forsiden kan bruger se om der er forbindelse mellem webapplikation og dronen, tilgå Flight bootcamp som er en quick-guiden til flyveopsætning og se en liste med et uddrag af de sidste flyvninger. Øverst på siden kan bruger tilgå opsætning af ny flyvning og den fulde liste med tidligere flyvninger.



Figur 4.2: Velkommen vindue

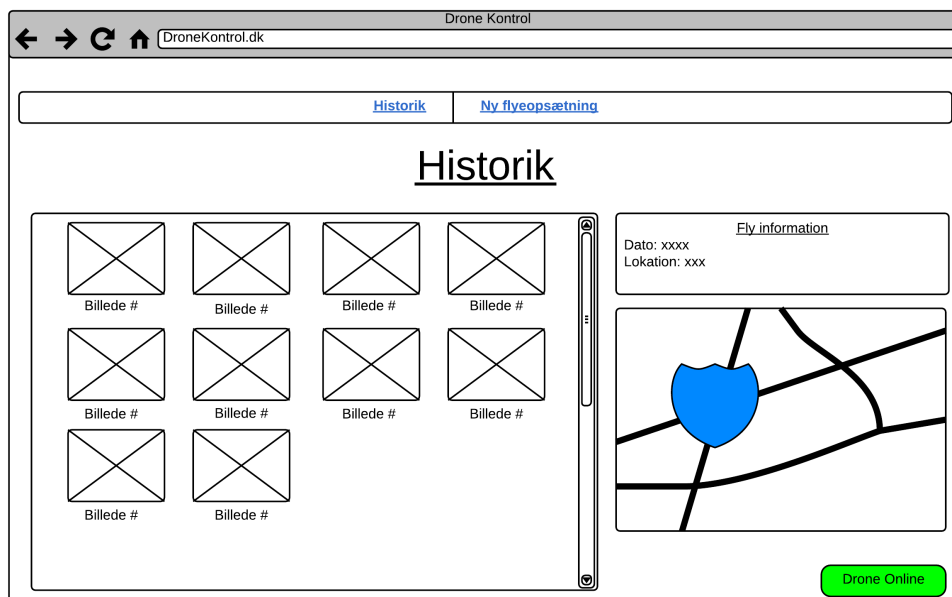
I historik menuen har bruger mulighed for at se alle tidligere flyvninger. De tidligere flyvninger præsenteres med hver sin mappe, og mapperne er struktureret efter dato. Da de tidligere flyvninger er struktureret efter dato er det nemt og hurtigt at finde data fra den eller de ønskede flyvninger. Når bruger doubleklikker på en tidligere flyvning åbnes tilhørende mappe, og alt indhold i mappen vises.



Figur 4.3: Historik vindue

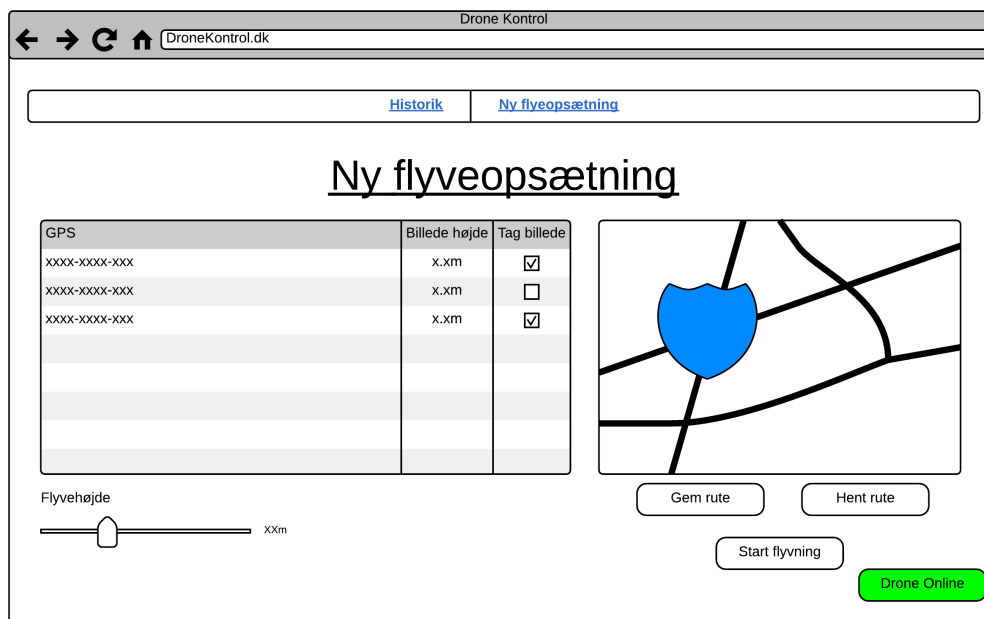


Når en tidligere flyvning er valgt, præsenteres bruger for al information tilknyttet den pågældende flyvning. Hvilket betyder bruger får adgang til flyverute, billeder og film sekvenser. På et kort vises den flyveruten dronen gjorde brug af.



**Figur 4.4:** Tidligere flyverute valgt

I Opsæt ny flyvning menuen kan bruger indstille ny flyveopsætning. Bruger kan indstille sin ønskede flyverute ved at klikke på kortet, og vælge hvilke GPS positioner som dronen skal overvåge og tage billeder af. Hver GPS position der vælges præsenteres i tabellen til venstre for kortet. I tabellen kan bruger ydermere indstille om der skal tages billeder ved GPS positionerne og hvilken højde dronen skal tage billeder fra. Nedenfor tabellen sættes den generelle flyvehøjde. Bruger har mulighed for at gemme nylavede flyveopsætninger til senere brug.



**Figur 4.5:** Ny flyverute