

# *Kravspekifikation*

---

## *Autonom overvågningsdrone*



---

Rasmus Lydixen  
11647

---

Anders Opstrup  
11726

---

Kevin Grooters  
11655



**Ingeniørhøjskolen**  
**Aarhus Universitet**  
Finlandsgade 22  
8200 Aarhus N

**Projekt titel:**

Autonom overvågningsdrone

**Projekt:**

Bachelorprojekt

**Projektperiode:**

August 2014 - December 2014

**Projektgruppe:**

14123

**Gruppemedlemmer:**

11647 – Rasmus T. Lydixsen  
11762 – Anders H. Opstrup  
11655 – Kevin Grooters

**Vejleder:**

Torben Gregersen

**Samlet sidetal: 15**

**Projekt afsluttet: 17-12-2014**

# 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	Brugssituation . . . . .	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 til projektet: *Autonom overvågningsdrone*. Krav og egenskaber for systemet beskrives i dokumentet.

## 1.4 Systembeskrivelse

På Ingeniørhøjskolen Aarhus Universitet forefindes en AeroQuad ARF Quadcopter. Målet med projektet er at omdanne quadcopteren til en autonom overvågningsdrone.

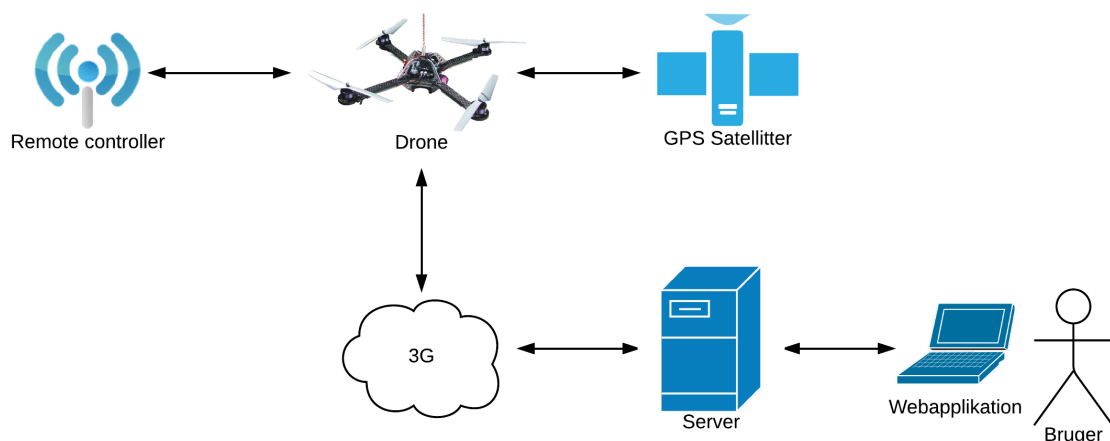
Dronen skal ud fra brugers anvisninger overvåge og tage billeder af et defineret område. Dronen tilgås via en webapplikation der fungerer som en grafisk brugerflade mellem bruger og server. Via webapplikationen kan bruger oprette flyveopsætninger til drone samt se billeder og flyverute fra tidligere flyvninger. Når der laves ny flyveopsætning vælger bruger en række GPS positioner som drone skal flyve til, flyvehøjde og hvorvidt der skal tages billeder ved de valgte GPS positioner. Når bruger har lavet en ny flyveopsætning, stilles flyveopsætningen tilgængelig for dronen på server.

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. Til enhver tid, skal kommunikation mellem drone og server foregå via mobilt netværk. Der gøres hovedsageligt brug af 3G netværket, men i områder med dårlig forbindelse vil der blive gjort brug af 2G som fall-back netværk.

## 1.5 Systemoversigt

Bruger benytter en computer til at tilgå webapplikation og lave en ny flyveopsætning. Når bruger har lavet en ny flyveopsætning, overføres flyveopsætningen via internettet til server, hvor den gøres tilgængelig for dronen.

Under flyvning kontrollerer drone løbende egen GPS position via kommunikation med GPS satellitter. Dette gør den for at opdatere egen position og for efterfølgende at kunne beregne den korrekte flyveorientering. Via det mobile 3G netværk kommunikerer drone med server. Dronen sender server information om nuværende GPS position, overfører billeder og kontrollerer om der er en ny flyveopsætning tilgængelig.



*Figur 1.1:* Systemskitse

## 1.6 Brugssituation

Et typisk brugsscenarie for den autonome overvågningsdrone er 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 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 højest, 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.

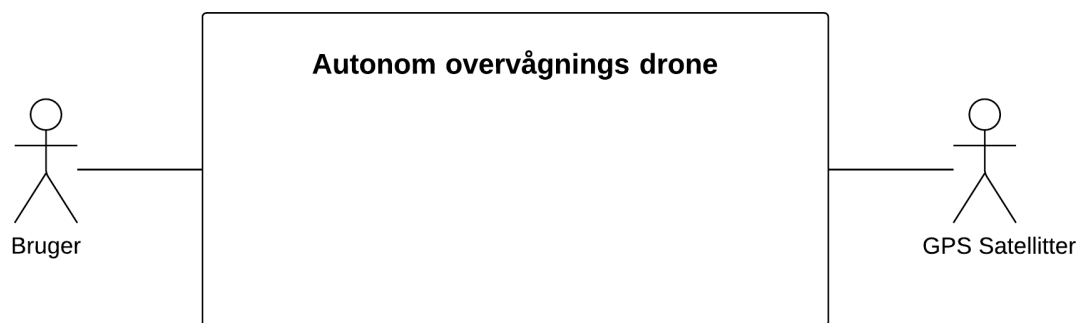
*Tabel 1.3:* Prioriteringsliste

# 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 2.1 viser hvilke aktører der interagerer med systemet.



*Figur 2.1:* Aktør diagram

## 2.2 Aktørbeskrivelser

Aktørbeskrivelsen skitserer systemets aktører samt hvilke 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.

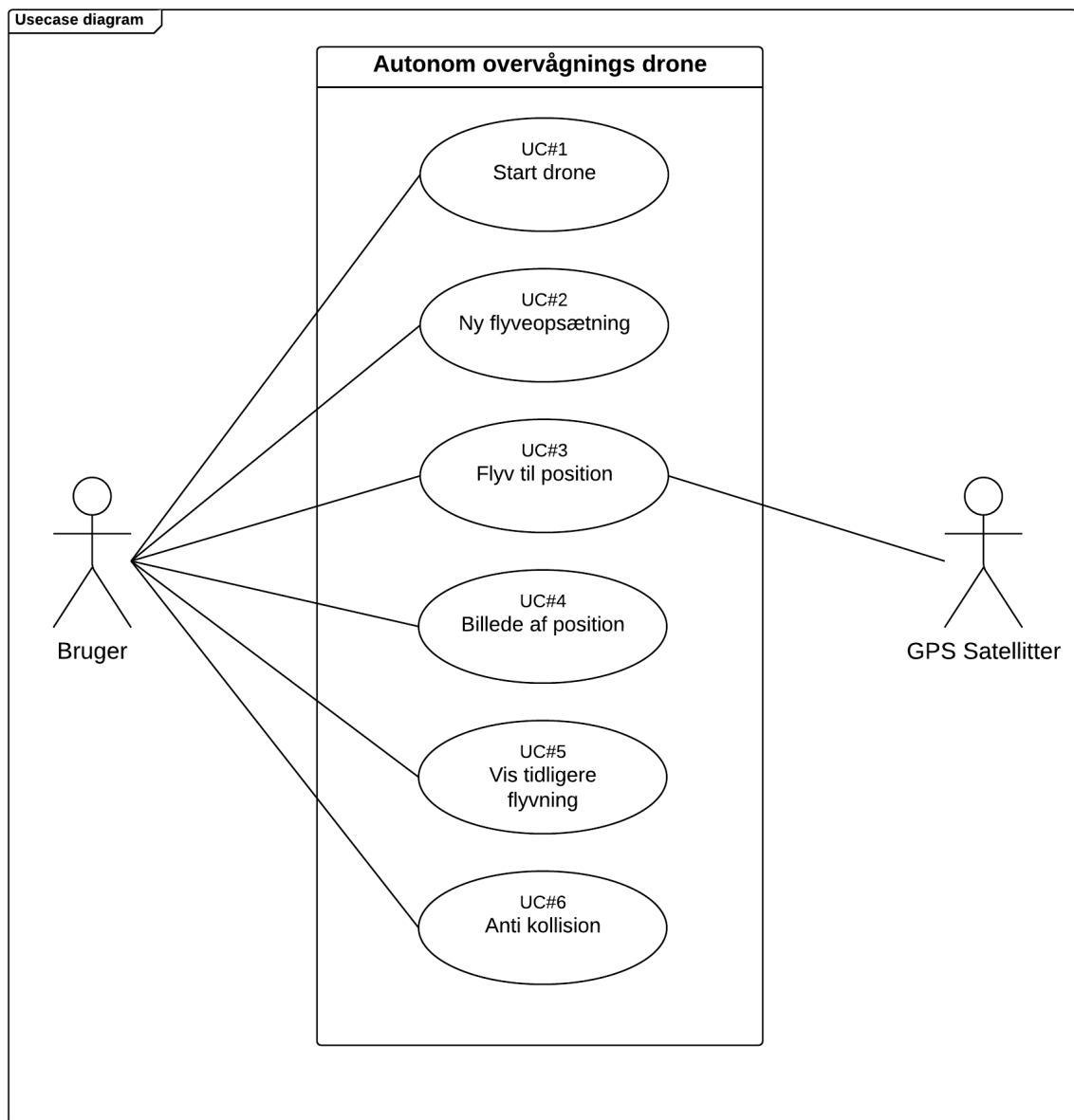
*Tabel 2.1:* Aktørbeskrivelse, Bruger

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 først i udviklingsforløbet er mest essentielle for systemet, mens de senere iterationer er mindre essentielle for systemet. Udviklingsforløbet er planlagt til at forløbe via 4 iterationer. Nedenfor beskrives indholdet af de forskellige iterationer.

### **Iteration 1:**

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

### **Iteration 2:**

I iteration 2 er hovedformålet at få styr på kommunikationen mellem server 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 drone 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 der tages under flyvning sendes via mobilnet fra dronen til server. Billederne gøres tilgængelige for bruger på webapplikationen. 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 dette afsnit vises fully dressed use case beskrivelser af use cases vist på figur 2.3.

### UC1 - Start drone

Mål	Drone er tændt og har forbindelse til webapplikation.
Initiering	Bruger.
Aktører og stakeholders	Bruger (primær aktør) <ul style="list-style-type: none"><li>• Bruger tænder drone.</li></ul>
Startbetingelser	Ingen.
Slutbetingelser	Drone kan modtage flyveopsætning fra server.
Hovedforløb	<ol style="list-style-type: none"><li>1. Bruger tilslutter batteri og drone 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.<ol style="list-style-type: none"><li>a: Forbindelse kan ikke oprettes.</li></ol></li><li>5. Drones nuværende GPS position sendes til server.</li></ol>
Undtagelser	<ol style="list-style-type: none"><li>a: Forbindelse kan ikke oprettes.<ul style="list-style-type: none"><li>• Systemet indikerer fejl og derfor genstarter bruger systemet.</li></ul></li></ol>

**Tabel 2.3:** Use Case 1

## UC2 - Ny flyveopsætning

Mål	Ny flyveopsætning er oprettet og tilgængelig for drone.
Initiering	Bruger.
Aktører og stakeholders	Bruger (primær aktør) <ul style="list-style-type: none"> <li>• Bruger ønsker at få adgang til webapplikation.</li> <li>• Fra webapplikation indstiller bruger flyveopsætning.</li> <li>• Bruger gør flyveopsætning tilgængelig på server.</li> </ul>
Startbetingelser	Bruger er oprettet i systemet og UC#1 er succesfuld gennemført
Slutbetingelser	Ny flyveopsætning er tilgængelig for drone.
Hovedforløb	<ol style="list-style-type: none"> <li>1. Bruger logger på webapplikation. <ol style="list-style-type: none"> <li>a: 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>b: Der laves ikke ny flyveopsætning.</li> </ol> </li> <li>4. Bruger gemmer flyveopsætning, og flyveopsætningen gøres tilgængelig på server.</li> </ol>
Undtagelser	<ol style="list-style-type: none"> <li>a: Fejl i login. <ul style="list-style-type: none"> <li>• Bruger føres tilbage til login.</li> </ul> </li> <li>b: Der laves ikke ny flyveopsætning. <ul style="list-style-type: none"> <li>• Gemt flyveopsætning benyttes.</li> </ul> </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 stakeholders	<p>Bruger (stakeholder)</p> <ul style="list-style-type: none"> <li>• Bruger ønsker at drone flyver som angivet i flyveopsætning.</li> </ul> <p>GPS satellitter (sekundær aktør)</p> <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. <ol style="list-style-type: none"> <li>a: Dronen henter ikke flyveopsætning.</li> </ol> </li> <li>2. Nuværende position opdateres. <ol style="list-style-type: none"> <li>b: Ugyldig GPS koordinat.</li> </ol> </li> <li>3. Flyvehøjde tilpasses. <ol style="list-style-type: none"> <li>c: Ugyldig flyvehøjde.</li> </ol> </li> <li>4. Flyveorientering tilpasses.</li> <li>5. Drone flyver mod ønsket position.</li> <li>6. Ønsket position er nået. <ol style="list-style-type: none"> <li>d: Ønsket position er ikke nået.</li> </ol> </li> </ol>
Undtagelser	<p>a: Dronen henter ikke flyveopsætning.</p> <ul style="list-style-type: none"> <li>• Flyvning med en anden flyveopsætning er aktiv.</li> </ul> <p>b: Ugyldig GPS koordinat.</p> <ul style="list-style-type: none"> <li>• Drone går i fejlmode #1.</li> </ul> <p>c: Ugyldig flyvehøjde.</p> <ul style="list-style-type: none"> <li>• Drone går i fejlmode #2.</li> </ul> <p>d: Ønsket position er ikke nået.</p> <ul style="list-style-type: none"> <li>• UC #3 genstartes.</li> </ul>

**Tabel 2.5:** Use Case 3

## UC4 - Billede af position

Mål	Drone tager et billede af nuværende position, og sender billedet til server. Fra webapplikation kan bruger se og acceptere billedet.
Initiering	UC#3.
Aktører og stakeholders	Bruger (primær aktør) <ul style="list-style-type: none"><li>• Inspicerer og accepterer billede.</li></ul>
Startbetingelser	UC#3 er succesfuld gennemført.
Slutbetingelser	Bruger kan tilgå billede via webapplikation.
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 billede via webapplikation.<ol style="list-style-type: none"><li>a: Bruger beder om et nyt billede.</li><li>b: Bruger svarer ikke inden for tidsgrænsen.</li></ol></li></ol>
Undtagelser	<ol style="list-style-type: none"><li>a: Bruger beder om et nyt billede.<ul style="list-style-type: none"><li>• Drone instrueres til at ændre højde, orientering eller position.</li></ul></li><li>b: Bruger svarer ikke inden for tidsgrænsen.<ul style="list-style-type: none"><li>• Drone får automatisk tildelt accept.</li></ul></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 stakeholders	Bruger (primær aktør) <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. <ol style="list-style-type: none"> <li>a: Fejl i login.</li> </ol> </li> <li>2. Fra forsiden navigerer bruger til historik.</li> <li>3. Bruger vælger en specifik tidligere flyvning.</li> </ol>
Undtagelser	<ol style="list-style-type: none"> <li>a: Fejl i login. <ul style="list-style-type: none"> <li>• Bruger føres tilbage til login skærm.</li> </ul> </li> </ol>

**Tabel 2.7:** Use Case 5

## UC6 - Anti kollision

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

**Tabel 2.8:** Use Case 6

# Ikke-funktionelle krav 3

---

De ikke-funktionelle krav er krav til timings, kommunikation og dataopsamling. Ikke-funktionelle inddeles i følgende 4 grupper: Generelle krav, krav til webapplikation, krav til drone og krav til dataopsamling.

## 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 server

- 2.1. Indholder database med billeder og flyveruter.
- 2.2. Indholder database med brugere.

## 3. Krav til webapplikation

- 3.1. Webapplikation skal kunne tilgås via både computere og telefoner.
- 3.2. Load time skal være under 0.5 sekunder.
- 3.3. Bruger laver flyveopsætning ved hjælp af kort.

## 4. Krav til dronen

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

## 5. Krav til opsamling af data

- 5.1. Tiden mellem et billede tages og til det er tilgængeligt på webapplikation skal maksimalt være 5 sekunder.
- 5.2. Gyldig højdemåling ligger i intervallet 0,5 til 4,5 meter.
- 5.3. 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

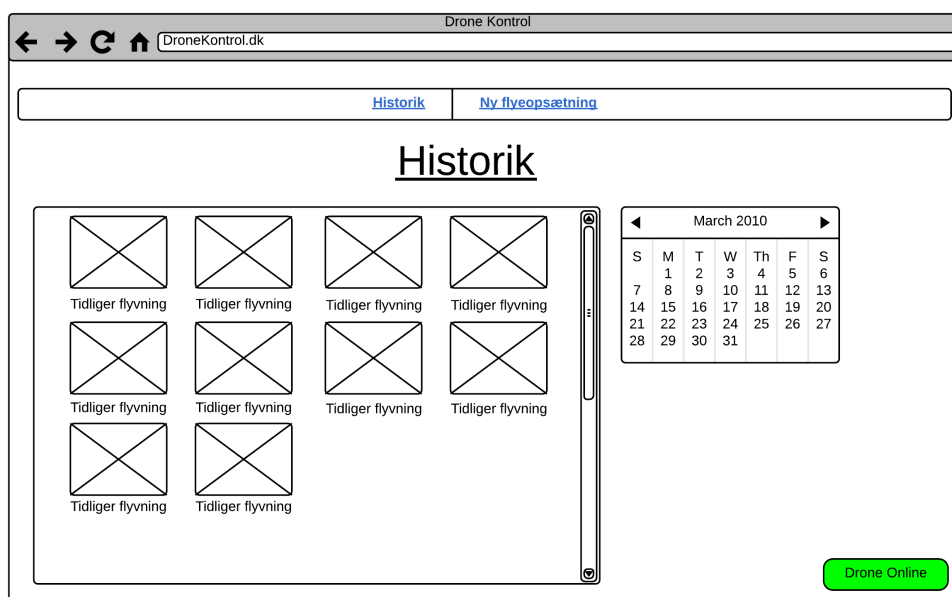


Efter login præsenteres bruger for webapplikations forside, se figur 4.2. Fra forsiden kan bruger se om der er forbindelse mellem webapplikation og dronen, tilgå Flight bootcamp som er en quick-guiden til flyveopsætning og se en liste med et uddrag af de sidste flyvninger. Øverst på siden kan bruger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



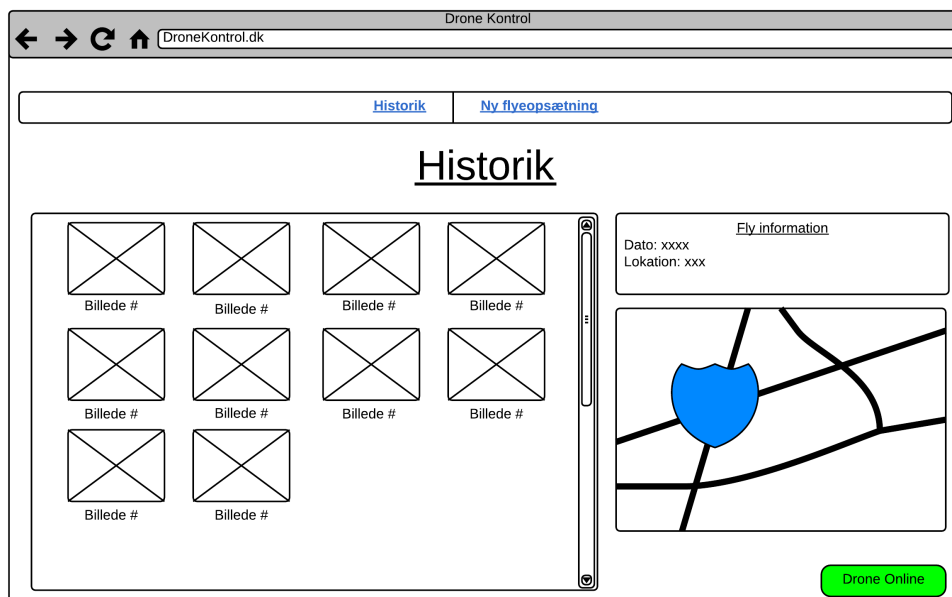
Figur 4.2: Velkommen vindue

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.



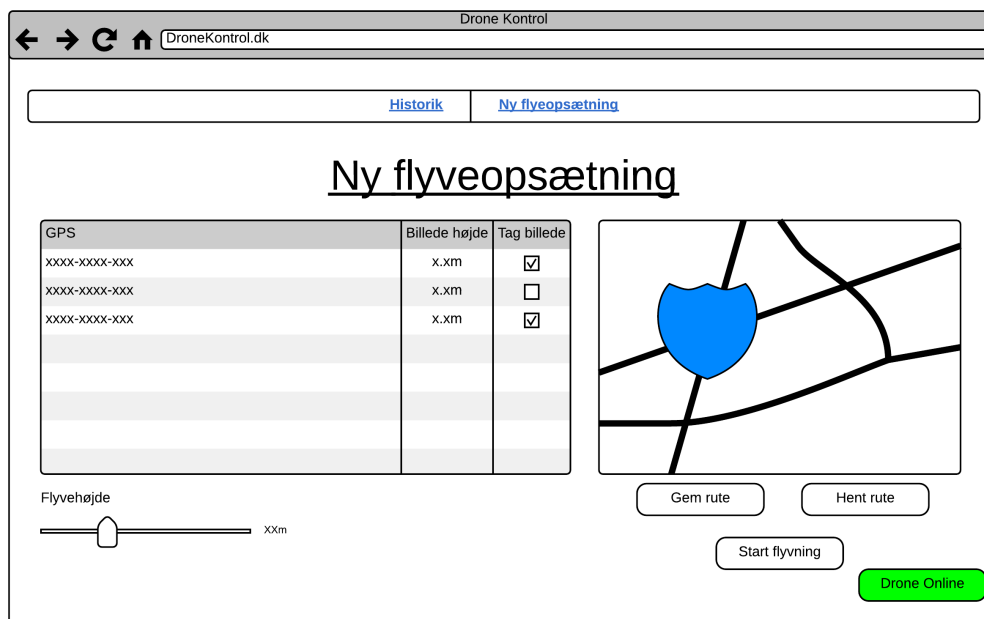
Figur 4.3: Historik vindue

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