

AARHUS UNIVERSITET  
SEMESTERPROJEKT 4  
GRUPPE 5

---

Accepttestspecifikation  
CarnGo

---

*Gruppemedlemmer:*

Edward Hestnes Brunton

(201705579)

Marcus Gasberg

(201709164)

Martin Gildberg Jespersen

(201706221)

Mathias Magnild Hansen

(201404884)

Tristan Moeller

(201706862)

Erik Mowinckel

(20107667)

Hamza Ben Abdallah

(201609060)

*Vejleder:*

Jesper Michael Kristensen, Lektor

29. maj 2019



# 1 Versionshistorik

Ver.	Dato	Initialer	Beskrivelse
1.0	3/04-2019	MH	Accepttest af ikke-funktionelle krav

# Indhold

<b>1</b>	<b>Versionshistorik</b>	<b>1</b>
<b>2</b>	<b>Accepttest</b>	<b>3</b>
2.0.1	Microsoft UI Automation . . . . .	3
2.1	Funktionelle krav . . . . .	3
2.2	Ikke funktionelle krav . . . . .	4

## 2 Accepttest

I dette afsnit beskrives accepttests for funktionelle og ikke-funktionelle krav. Accepttests viser om produktet opfylder de krav, som der blev stillet i kravspecifikationen. Siden der er blevet benyttet MoSCoW prioritering i ikke funktionelle krav, er der nogle krav som har højere prioritet, og dermed er det vigtigere, at de bliver godkendt[1]. Testene er udført d. 24/05/2019. De funktionelle krav testes ved hjælp af Coded UI[2], hvor de ikke funktionelle krav bliver fortaget enten manuelt af en person eller af Coded UI.

### 2.0.1 Microsoft UI Automation

Microsoft UI Automation (UIA) er et API, som gør det muligt at identificere og manipulere applikationens brugergrænsefladeelementer- og objekter[2]. Dette værktøj anvendes til at automatisere tests for både funktionelle og ikke-funktionelle krav. Her angives specifikke instruktioner til, hvordan brugergrænsefladen skal manipuleres, baseret på accepttestspecifikationerne angivet for hver User Story og enkelte ikke-funktionelle krav. Her opstilles forsikringer om grænsefladens respons til brugeren. Hvis UIA ikke registrerer respons i overensstemmelse med forsikringerne, vil testene fejle. Denne form for GUI-testing tester hele systemet og interagerer med brugergrænsefladen svarende til, hvad en testperson ville gøre manuelt. Dette giver en effektiv form for feedback, idet hver gang der opdateres i applikationens kildekode, vil accepttestene blive kørt. Hvis de fejler, notificeres gruppemedlemmerne med det samme, som beskrevet under Continuous Integration i bilag **TestDokument**.

Strukturen af applikationen gør, at det ikke altid er muligt at hævde på de konkrete grafiske elementer. Her må der anvendes White Box testning, hvor der assertes på elementer i businesslogikken frem for brugergrænsefladen.

Nedenfor ses resultaterne for accepttesten for de funktionelle krav. Det fremgår, at de fleste af kravene er godkendt, hvilket kan bekræftes af CodedUI tests. Der er ni tests, som er godkendt ud af ti. Fjernelse af brugerprofil er ikke implementeret, hvorfor den ikke er godkendt. Det vurderes, at denne funktionalitet har mindst værdi for de personas, som er beskrevet tidligere. Derfor har den haft lavest prioritet.

### 2.1 Funktionelle krav

Testgrundlag	Beskrivelse	Status
US 1	Oprettelse af brugerprofil	Godkendt
US 2	Redigering af brugerprofil	Godkendt
US 3	Oprettelse af bilprofil	Godkendt
US 4	Fjernelse af bilprofil	Godkendt
US 5	Redigering af bilprofil	Godkendt
US 6	Søgning efter biler til leje	Godkendt
US 7	Anmodning om biludleje	Godkendt
US 8	Notifikation til lejer om svar på anmodning om leje af bil	Godkendt
US 9	Login på applikation	Godkendt
US 10	Fjernelse af brugerprofil	Ikke godkendt

Tabel 2.1: Accepttest resultat for funktionelle krav

2.2 Ikke funktionelle krav

Functionality

Krav ID	Krav	Test	Forventet Resulat	Faktisk Resultat	Vurdering (OK/FAIL)
K1.1	CarnGo applikationen skal aldrig fryse, dvs. ophøre med at respondere på brugerens input, i mere end 3 sekunder af gangen	10 forskellige brugere benytter applikationen på samme tid, og det registreres at responstiden på de forskellige input er under tre sekunder	Responstiden er under tre sekunder	Responstiden er under tre sekunder	OK
K1.2	CarnGos knapper skal reagere, når der bliver sluppet på knappen	Trykker på knapper i applikationen	Når man giver slip på en knap i applikationen udføres funktionaliteten	Applikationen reagerer, når der bliver sluppet på en knap.	OK
K1.3	CarnGo skal anvende GPS koordinater, så brugere kan leje biler baseret på deres lokalitet	Ikke implementert	Ikke testet	N/A	N/A
K1.4	CarnGo skal give brugere mulighed for, at kommunikere indbyrdes gennem en chat tjeneste	Tryk på rent car og derved send en besked, derefter se om udlejeren får en besked.	Udlejeren får en besked	Udlejeren fik en besked	OK

Usability

Krav ID	Krav	Test	Forventet Resulat	Faktisk Resultat	Vurdering (OK/FAIL)
K2.1	CarnGo applikationen skal have en blå farve som hovedtema med farvekoden: #397be5	Der bliver brugt Coded UI til at teste det	Fargen avleses til #397be5	Farven læses til det rigtige	OK
K2.2	CarnGo applikationen skal have en hvid farve som baggrundsfarve med farvekoden: #fafafa	Der bliver brugt Coded UI til at teste det	Fargen avleses til #fafafa	Farven læses til det rigtige	OK
K2.3	CarnGo applikationen skal have en grå farve som forgrundsfarve med farvekoden: #686868	Der bliver brugt Coded UI til at teste det	Fargen avleses til #686868	Farven læses til det rigtige	OK
K2.4	CarnGos typografi skal ha et kontrastforhold over 4.5:1 for å følge WCAG AA standarden	Der bliver brugt Coded UI til at teste det	Begregningen viser et kontrastforhold av 4.5:1 eller over	Kontrastforholdet bliver udregnet til at passe	OK
K2.5	CarnGo skal kun have typografitypen Roboto	Typografien i CarnGo sammenlignes med Roboto	Typografien i CarnGo stemmer overens med Roboto	Typografien i CarnGo stemmer overens med Roboto	OK

Krav ID	Krav	Test	Forventet Resulat	Faktisk Resultat	Vurdering (OK/FAIL)
K2.6	CarnGo's typografi skal ikke have en punktstørrelse mindre end 10	Coded UI tester dette	Coded UI siger at ingen typografi er under punktstørrelsen 10	Coded UI testen kom tilbage uden fejl	OK
K2.7	CarnGo's typografi skal ikke have en punktstørrelse større end 50	Coded UI tester dette	Coded UI siger at ingen typografi er over punktstørrelsen 30	Coded Ui kom tilbage uden fejl	OK
K2.8	CarnGo's GUI elementer skal skalere uniformt	Størrelsen på CarnGo vinduet endres	GUI elementene skalerer uniformt	GUI elementerne skalerer uniformt	OK

Reliability

Krav ID	Krav	Test	Forventet Resulat	Faktisk Resultat	Vurdering (OK/FAIL)
K3.1	CarnGo skal håndtere 10 aktive brugere af applikationen på samme tid	CarnGo applikationen bliver logget ind på med 10 forskellige brugere på samme tid	Alle brugere kan logge ind	Alle brugere kunne logge ind	OK
K3.2	CarnGo skal håndtere 20 inaktive brugere af applikationen på samme tid	Det logges inn på CarnGo på 20 forskellige brukere på samme tid	Alle brukerne opprettholder kontakten med CarnGo serveren	Alle bruger kunne være på applikationen inaktivt	OK



Performance

Krav ID	Krav	Test	Forventet Resultat	Faktisk Resultat	Vurdering (OK/FAIL)
K4.1	CarnGo skal være klar til brug inden for 10 sekund efter, at applikationen åbnes	Det spilles inn en video hvor CarnGo åpnes, videoen avleses for å se åpnings tiden	Åpnings tiden er under 10 sekunder	Åbningstiden er under 10 sekunder	OK
K4.2	CarnGo skal give respons inden for 10 sekunder, når en søgning foretages	Det spilles inn en video hvor brukeren søker på en bil i CarnGo, videoen avleses for å se respons tiden	Respons tiden er under 10 sekunder	Responstiden er under 10 sekunder	OK

Maintainability

Krav ID	Krav	Test	Forventet Resulat	Faktisk Resultat	Vurdering (OK/FAIL)
K5.1	CarnGo's GUI kode skal følge et GUI design pattern, der gør det nemt at overføre til andre platforme	Ikke testbart	N/A	N/A	N/A
K5.2	CarnGo skal kunne tilgås på en PC med internet forbindelse og Windows 10 OS	CarnGo åpnes på en PC om kjører Windows 10 og logger inn	Applikasjonen åpner og brukeren blir logget inn	Applikationen virker	OK
K5.3	CarnGo's typografitype skal være gratis	Ikke testbart	N/A	N/A	N/A

Security

Krav ID	Krav	Test	Forventet Resulat	Faktisk Resultat	Vurdering (OK/FAIL)
K6.1	CarnGo skal censurere brugerens password, når der logges ind på applikationen	Det inntastes et password i password feltet på login siden	Passwordet sensureres	Passwordboksen er censureret	OK
K6.2	CarnGo skal anvende kryptering, ved behandling af brugernes data	Ikke implementert	Ikke testet	N/A	N/A

Referencer

[1] Wikipedia, „Moscow Method“, 2019. side: [https://en.wikipedia.org/wiki/MoSCoW\\_method](https://en.wikipedia.org/wiki/MoSCoW_method).

[2] „Coded UI Documentation“, tek. rap., 2019, s. 1–9. DOI: .1037//0033-2909.I26.1.78. side: <https://docs.microsoft.com/en-us/aspnet/identity/overview/>.