

Newton

# Projektuppgift

## **Planera, genomföra och följa upp test (fortsättning)**

### **Grupp 2**

Kurs: Planera, genomföra och följa upp test (fortsättning)

Termin och år: VT-2019

Författare: Renc, Rizgar, Kevin

Lärare: Olof Thoors

## Innehållsförteckning

<b>Innehållsförteckning</b>	<b>2</b>
<b>Sammanfattning</b>	<b>3</b>
<b>Nyckelord</b>	<b>3</b>
<b>Abstract</b>	<b>4</b>
<b>Inledning</b>	<b>5</b>
<b>Projektmedlemmar</b>	<b>5</b>
<b>Genomförande: metodik, teknik</b>	<b>6</b>
<b>Testmetoder</b>	<b>7</b>
<b>Genomförande av test</b>	<b>8</b>
<b>Defekter</b>	<b>12</b>
<b>Systemmiljö</b>	<b>15</b>
<b>Testmiljö</b>	<b>16</b>
<b>Projektanalys</b>	<b>17</b>
<b>Produktanalys</b>	<b>19</b>
<b>Testfokusanalys</b>	<b>20</b>
<b>Root Cause Analysis</b>	<b>21</b>
<b>Tidsplan</b>	<b>23</b>
<b>Rekommendationer</b>	<b>24</b>
<b>Bilagor</b>	<b>25</b>
<b>Referenslista</b>	<b>26</b>

## Sammanfattning

Projektet handlar om att testa en nyutvecklad hemsida som är byggd i Wordpress. Under projektets gång har vi genomfört projektanalys och tillämpat olika verktyg, metoder, tekniker, modeller för att testa och kvalitetssäkra funktionaliteten på Conil. Detta dokument innehåller mål, omfattning, schema, risker, testverktyg tillvägagångssätt, projektmedlemmar, defekter, systemmiljö, testmiljö och root cause analysis. Det här dokumentet kommer tydligt att identifiera hur vi följt upp projektarbetet samt vad testresultaten kommer att vara.

## Nyckelord

*Funktionalitet, Projekt Conil, Testning, Kvalitetssäkring, Testmetoder.*

## **Abstract**

The project is about testing a newly developed website that is built in Wordpress. During the project, we have carried out risk analyzes, risk identification and applied various tools, methods, techniques, models for testing and quality-assured functionality on Conil. This document contains goals, scope, schedule, risks, test tool approaches, project members, defects, system environment, test environment and root cause analysis. This document will clearly identify how we followed up the project as well as what the test results will be.

## Inledning

I denna rapport följer vi upp utvecklingsprocessen för att testa funktionaliteten på hemsidan ([www.conil.se](http://www.conil.se)). Detta dokument ger en översikt över vår testplan samt information om alla delar i planering, testfas och utförandet av tester. Syftet med den här projektet är att bli introducerade till de moment som ingår i vårt framtida yrke.

### 1.1 Syfte

Huvudsyftet är att som blivande testare hitta fel och hjälpa till att testa och kvalitetsgranska hemsidan conil.se och se till att felen åtgärdas innan leverans till kunden. Detta för att förhindra att kunden blir missnöjd och förlorar förtroende för oss. Kvalitetsgranskningen sker i takt med varje iteration av hemsidans utveckling och vi kommer att regressionstesta efter varje iteration för att garantera kvaliteten håller våra krav. Dessutom har vi ansvar över att identifiera möjliga förbättringsområden som kommer att levereras med den slutgiltiga produkten.

### 1.2 Mål

Försäkra att systemet uppfyller kravet. Vid slutet av projektutvecklingscykeln ska användaren hitta att projektet har uppfyllt eller överträffat alla sina förväntningar enligt detaljerna i kraven.

### 1.3 Primära mål

Primära målet med Conil kommer vara bland annat att:

- Identifiera och avslöja problem och tillhörande risker
- Meddela alla kända problem till projektgruppen och se till att alla problem rapporteras i JIRA.

### 1.4 Projektmedlemmar

Namn	Ansvarsområde
Renc Sadiki	Student - testare
Rizgar Teimouri	Student - testare
Kevin Zhang	Student - testare

## 2 Genomförande: metodik, teknik

- *Här beskrivs de metoder som användes för det faktiska test-genomförandet av projektet. Det syftar på hur projektet testades, planerades och strukturerades.*

I projektet har vi jobbat i sprintar där varje sprint har pågått i en vecka. Vi har träffats i skolan och arbetat tre dagar i veckan. Varje arbetsdag har vi börjat med ett möte där vi diskuterat vad vi gjort och vad som behöver göras. De dagar vi inte har kunnat närvara i skolan har vi använt Discord för att ha möten på distans.

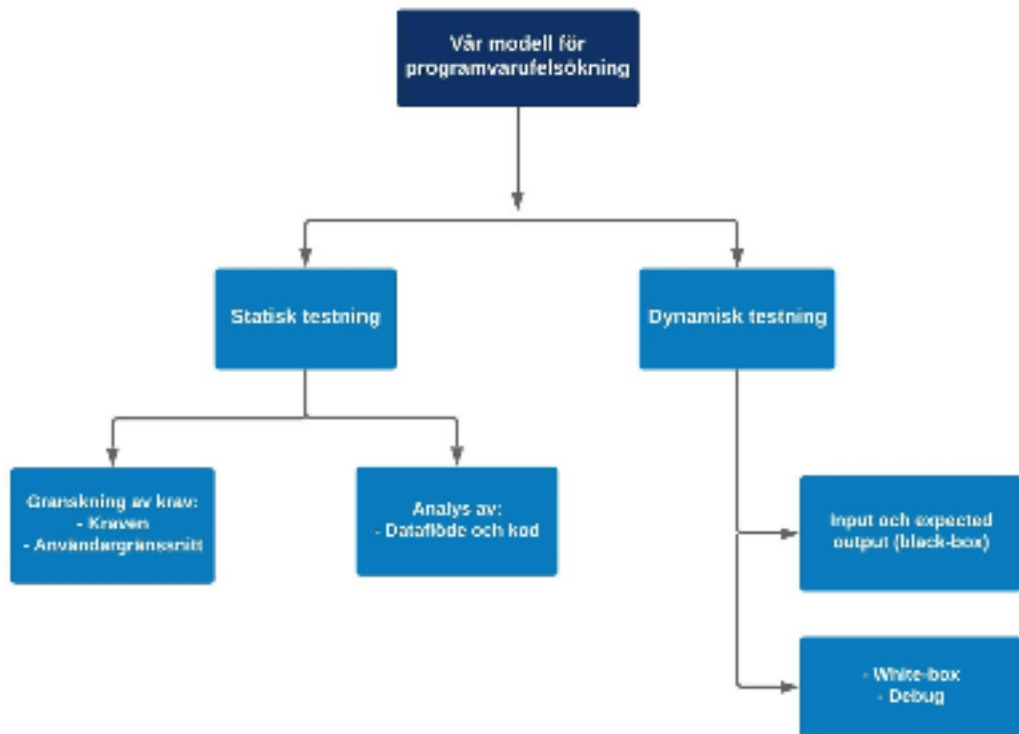
Under projektets gång har vi haft möten där vi gemensamt gått igenom projektet. Våra möten har varit tillfällen då vi har visat upp projektet så långt vi har kommit och läraren har haft möjlighet att ge återkoppling vad de tycker är bra och vad de tycker att vi ska lägga mer fokus på. Vi har haft kontakt med handledare varje vecka där vi har kunnat ställa frågor om projektet, projektdokumentation osv.

För att dokumentera all dokumentation kring projektet har vi använt Google Docs. Att använda Google Docs gjorde det enkelt för alla i gruppen att kunna bidra till dokumentationen.

När vi upptäckt något fel eller något som måste göras har vi skapat issues på JIRA. Issues har fungerat som en todo-lista där varje gruppmedlem har kunnat tilldela sig själv uppgifter. Issues gav en bra översikt av vad som behöver göras, det har varit enkelt för varje gruppmedlem att välja en uppgift att göra. Genom att rapportera buggar var det också möjligt att se vad alla i gruppen höll på med vilket minskade risken att vi gjorde dubbelarbete.

## 3 Testmetoder

Våra testmetoder visar hur testning gått till där vi delat in det i två kategorier: statiska och dynamiska tester (*se figur nedan*). Statisk testning fokuserar på att undersöka och granska programkod, dokumentation och design. Dynamiska tester är indelade i "black box"- och "white box" tester, vilket har gett oss möjligheten att upptäcka fel i tidigt skede.



## 3.1 Genomförande av test

### 3.2 Prestandatest

Prestandatest är något som används när det finns en osäkerhet om hur mycket arbete som mjukvaran klarar av utan att "ge upp". Syftet var att lasttesta webbsidan.

Vi utförde prestandatester på datorer med operativsystemen Windows 10 och macOS.

Dessa gjordes lokalt på datorn (Localhost/wordpress/) och på onlineversionen (Conil.se).

Anledningen till att vi gjorde testerna var för att utöka våra kunskaper om prestandatester samt bli mer bekanta med **Jmeter**.

- Localhost/wordpress och Conil.se testad med **Apache Jmeter** (version 5.1.1).

Label	#Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/s
HTTP Request	5	1097	1015	1252	93,01	0,00%	58,2/min	109,12
/#Casa Helena	5	1164	1020	1334	116,23	0,00%	58,1/min	108,93
/#Gallery	5	1183	1002	1349	135,49	0,00%	56,2/min	105,42
/#Location	5	1207	1122	1361	85,02	0,00%	54,8/min	102,72
/#Contact	5	1181	1101	1301	71,04	0,00%	54,8/min	102,82
/#English page	5	1606	1449	1767	117,35	0,00%	51,6/min	90,29
/#About Conil	5	1063	926	1292	126,87	0,00%	1,0/sec	113,07
/#Prices	5	1009	922	1146	74,95	0,00%	1,0/sec	114,85
Total	40	1189	922	1767	200,02	0,00%	3,0/sec	334,16

### Förklaring

<b>Label</b>	URL för den specifika HTTP förfrågan.
<b>Samples</b>	Antal virtuella anslutningar per förfrågan.
<b>Average</b>	Genomsnittlig tid för virtuella anslutningar för den specifika HTTP förfrågan.
<b>Min</b>	Minimal responstid.
<b>Max</b>	Maximal responstid.
<b>Std. Dev</b>	Standardavvikelse.
<b>Error%</b>	Antal defekter i procent.
<b>Throughput</b>	Antal förfrågningar servern hanterar per minut/sekund.



## 3.3 Funktionstest

Under funktionstestet har varje issue nedtecknats och rapporterats på JIRA. Varje "defekt" har fått ett specifikt id och till varje issue har information och kommentarer lagts. Information som behövs för att kunna identifiera och rätta till felet är bla. unik ID, mjukvarutyp, namn på det som testat, testfallsnummer och en kommentar om defekten.

## 3.4 Systemtest

Syftet med systemtest är att testa enligt de krav som är ställda på systemet är uppfyllda och om systemet beter sig som förväntat under olika förutsättningar. Det görs som black box testing.

# Newton

## 3.5 Enhetstest

Testscriptet skapades med hjälp av Selenium framework, TestNG och Eclipse.  
Sammanfattningsvis gick automatiseringen ut på att hitta alla aktiva/döda länkar på hemsidan. **Notering: Grön text = kommentar**

- Scriptet som implemtenterades finner vi nedan:

```
import java.util.ArrayList;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class Conil {
public static void main(String[] args) throws InterruptedException {

System.setProperty("webdriver.chrome.driver", "/Users/Användare/Downloads/chromedriver");
WebDriver driver = new ChromeDriver(); // chrome startas
driver.manage().window().maximize(); // helskrämsläge
driver.manage().deleteAllCookies(); // raderar kakor

// väntar
driver.manage().timeouts().pageLoadTimeout(40, TimeUnit.SECONDS);
driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
driver.get("https://localhost/wordpress/"); //ange URL

List<WebElement> linkslst = driver.findElements(By.tagName("a")); // hittar alla länkar

System.out.println("Full links " + linkslst.size()); //antal fungerande länkar
List<WebElement> activateLinks = new ArrayList<WebElement>(); // 450

for (int i = 0; i < linkslst.size(); i++) {
System.out.println(linkslst.get(i).getAttribute("href"));
if (linkslst.get(i).getAttribute("href") != null
&& (!linkslst.get(i).getAttribute("href").contains("javascript"))) {
activateLinks.add(linkslst.get(i));

System.out.println("Active links " + activateLinks.size()); //antal länkar
System.out.println("Dead links " + deadLinks.size()); //antal döda länkar
for (int j = 0; j < activateLinks.size(); j++) {
}
}
}
```

[Output]:

Full links: 65 //antal hittade länkar (totalt)

Active links: 45 //antal hittade fungerande länkar

Dead links: 20 //antal hittade döda länkar

## 4 Defekter

Vi gjorde exploratory testing på hemsidan. Både öppna och stängda buggar rapporterades i JIRA med information som status, beskrivning, screenshots/video, prioritet och severity.

Öppna buggar: **14**

Prioritet Severity	1 - High	2 - Major	3 - Minor	4 - No impact
1 - Systemstoppande				
2 - Allvarligt	63	38, 51, 59, 62		
3 - Mindre allvarligt		40, 54	43, 56	61
4 - Irriterande			48, 59	58, 60

*(Siffrorna i rutorna är Bugg-ID från Jira)*

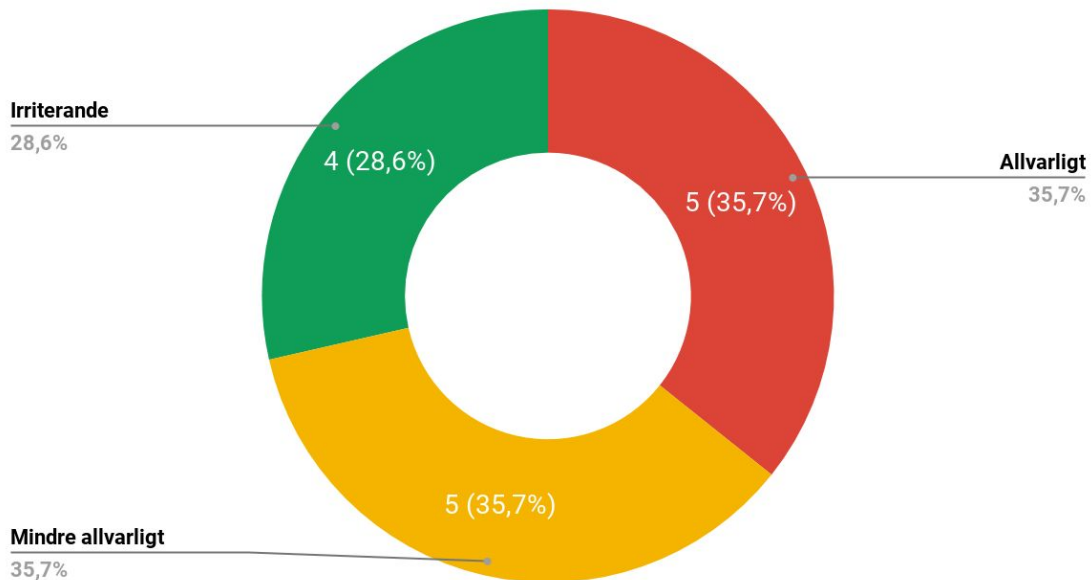
Stängda buggar: **27**

Prioritet Severity	1 - High	2 - Major	3 - Minor	4 - No impact
1 - Systemstoppande	55			
2 - Allvarligt	19, 23, 36, 41, 42, 44, 53	20, 37, 47, 50	49	18
3 - Mindre allvarligt	35		16	46
4 - Irriterande	9	21, 39	33, 43, 45, 52, 57, 64	17

*(Siffrorna i rutorna är Bugg-ID från Jira)*

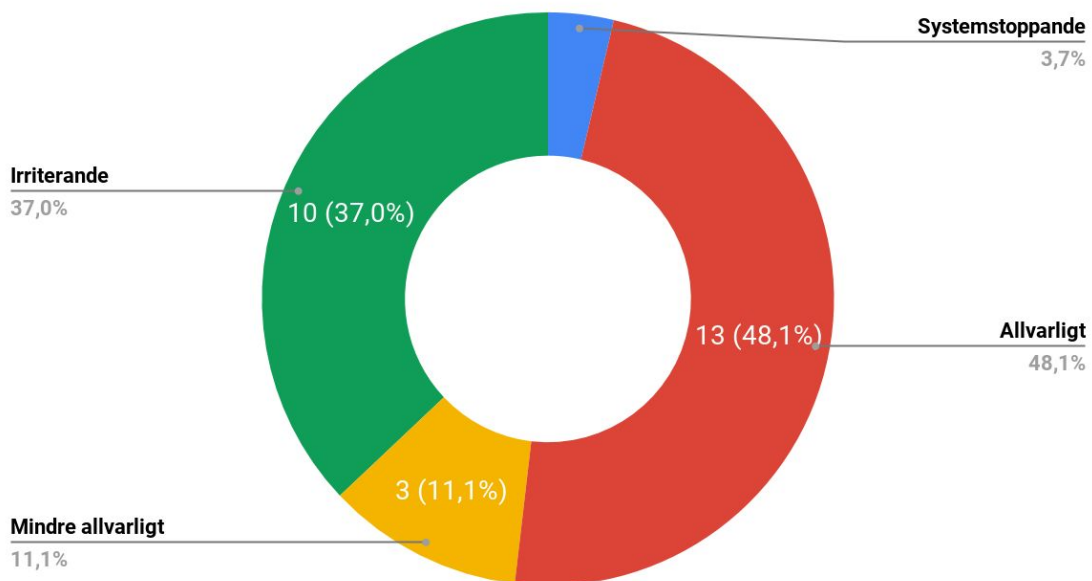
# Newton

## Antal öppna buggar



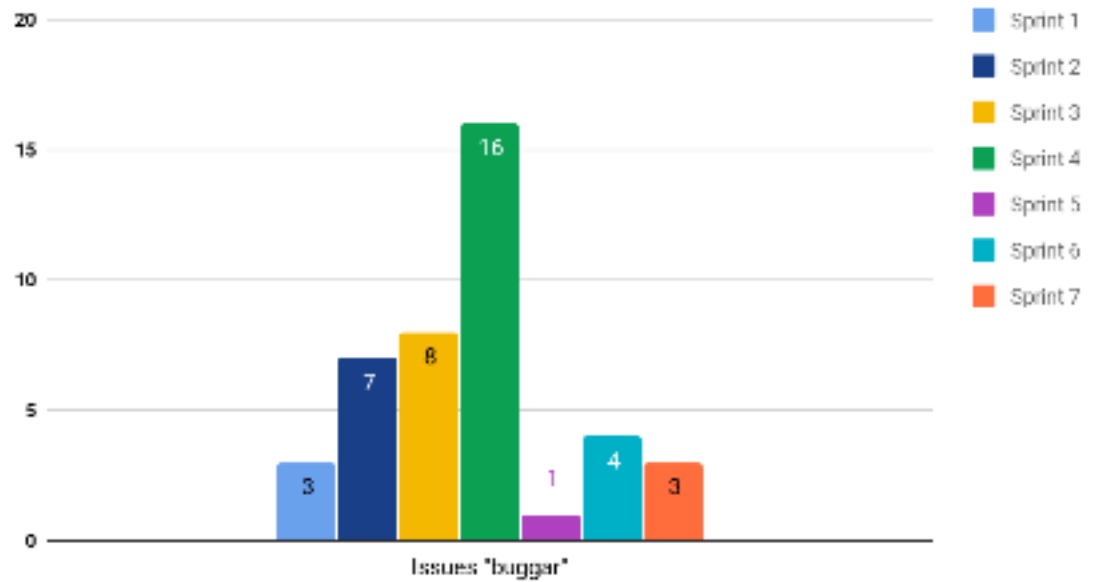
- Cirkeldiagrammet visar antal öppna/stängda buggar från sprint 1-7 där varje sprint varade i en vecka. Efter avslutad sprint sammanställde vi all resultat från JIRA.

## Antal stängda buggar



# Newton

Antal rapporterade buggar per sprint



Stapeldiagrammet visar totala resultatet från alla upphittade defekter från sprint 1-7 där vi sammanställde all resultat från Jira.

## 5 Systemmiljö

Följande verktyg användes eftersom de var open source verktyg, de var relativt enkla att hantera som nybörjare. Det var enkelt att följa upp verktygen med hjälp av tutorials på nätet.

- Under projektets gång har följande verktyg används:

Verktyg	Beskrivning
Jira	Verktyg för rapportering av buggar.
Trello	Verktyg för logga händelser i tavla.
Eclipse IDE	Programmeringsverktyg (Java).
Apache Jmeter	Verktyg för stresstester.
Discord	Verktyg för kommunikation.
Xdebug	Plugin integrering med Eclipse för debugläge.
Blazemeter	Cloud verktyg för prestandatester.
TestNG	Plugin integrering med Eclipse för Junit/enhetstester.
Selenium Framework	Verktyg för automatisering.

## 6 Testmiljö

Testmiljön är en uppsättning av mjukvara och hårdvara där testarna exekverar testfall. Vi använde oss av dessa testmetoder eftersom vi tyckte dessa metoder kunde ge oss bättre förståelse om vad det innebär att använda dessa metoder.

- Följande tabell dokumenterar testmiljökriterierna för varje testuppgift:

Testuppgift	Operativsystem	Databasnamn	Plattform
Stresstest	Mac OS X / Windows 10	localhost/wordpress	Apache Jmeter (version 5.1.1)
Enhetstest	Mac OS X / Windows 10	localhost/wordpress	Selenium + TestNG framework (Eclipse)
Prestandatest	Mac OS X / Windows 10	localhost/wordpress	Apache Jmeter (version 5.1.1)
Funktionstest	Mac OS X / Windows 10	localhost/wordpress	Safari (version 12.1)  Microsoft Edge (version 42.171)  Chrome (version 73.0.368)
Exploratory testing	Mac OS X / Windows 10	localhost/wordpress	Safari (version 12.1)  Microsoft Edge (Version 42.171)  Chrome (Version 73.0.368)



## 7 Projektanalys

### 7.1 Riskspecifikationer

Spalterna sannolikhet, konsekvens och prioritet berättar hur hög chans det är att risken kommer inträffa, hur stor konsekvens risken kommer att ge samt hur mycket det ska prioriteras. Det finns fem alternativ (1-5) där 1 är lägst och 5 är högst. Prioritetsspalten är ett resultat av sannolikhet och konsekvens.

- *Prioritet = Sannolikhet och Konsekvens.*

ID	Risk	Beskrivning	Sannolikhet	Konsekvens	Prioritet
1	Sjukdom	Sjukdomar kan förekomma inom projektgruppen som kan förhindra att man lyckas med projektets uppgifter.	5	1	2
2	Datorfel	Då vi jobbar med datorer finns det en risk för att programvara krånglar eller inte funkar som det ska.	5	3	4
4	Tidsplanering	Allt från många röda dagar nu på våren till att projektet är under tidspress.	5	2	3
5	Kommunikation	Då man kan lätt missuppfatta eller är missnöjd med någon annans insats etc.	3	5	5
6	Deadline	Inte fixa dokumentationen i tid.	1	4	5

## 7.2 Tekniska Riskspecifikationer

ID	Risk	Beskrivning	Sannolikhet	Konsekvens	Prioritet
7	Xdebug plugin	Xdebug plugin till Eclipse PHP är något vi laddat ner som ska vara till hjälp för att debugga webbservern conil men som vi inte har 100% kontroll över. Därför kan det uppstå hinder då plugin tillägget inte beter sig som man tänkt.	5	3	5
8	Internetuppkoppling	Då vi ständigt behöver vara uppkopplad till Internet för att ha tillgång till WordPress sidan kan vi inte göra så mycket om uppkopplingen ligger nere.	5	5	5
9	Apache Jmeter	Att kontrollera / bevisa att systemet kan hantera den förväntade belastningen; <ul style="list-style-type: none"> <li>- Att bestämma en systemkomponent som orsakar prestandadämpning.</li> <li>- Att jämföra kapacitet för två eller flera system/miljöer.</li> </ul>	5	5	2

## 8 Produktanalys

Vi delade upp projektet i 3 delar; område, sannolikhet och konsekvens.  
Siffrorna är baserade på testfokusanalysen under nästkommande rubrik.

En etta representerar låg sannolikhet för att ett fel inträffar respektive mild konsekvens. En trea representerar stor sannolikhet för att fel inträffar respektive allvarlig konsekvens.

Område	Sannolikhet	Konsekvens
1. Designfel	2 - Problem i andra enheter kan lätt uppstå då man använder exempelvis mobil eller Linux/Mac	3 - Svårtydlig text och man letar efter andra alternativ
2. Kontaktformulär	3 - Mail krävs för att påbörja bokning	3 - Går att boka tid men ej skicka mail
3. Bokningskalender	3 - Komplicerad funktion	3 - Viktigaste funktionen på sidan
4. Google Maps	3 - Fel på inbäddning av kod	1 - Går att zooma in och ut på kartor men inget utöver det
6. Information om området	2 - < >	1 - < >
7. Galleri	1 - Inbyggd Themify plugin i wordpress	2 - < >
8. Navigeringsbar	3 - < >	2 - < >
9. Stavfel	3 - Stavfel hos utvecklaren	2 - Går att läsa men det är irriterande med eventuella stavfel och siten tas oseriöst

## 9 Testfokusanalys

Målet med testfokusanalysen är att man tillsammans med övriga intressenter i projektet skapar en gemensam syn på prioriteringen av testerna. Med hjälp av analysen kunde vi rangordna testarbetet i sprintar för att säkerställa att de högst prioriterade områdena fick den tid som behövdes för att åtgärda defekterna.

Område:

1. Designfel
2. Kontaktformulär
3. Bokningskalender
4. Google Maps
5. Information om området
6. Galleri
7. Navigeringsbar
8. Stavfel

### - Riskpoints

	- Hög risk	Bugg som påverkar systemet allvarligt och kräver åtgärkning så tidigt som möjligt
	- Medium risk	Det innebär en bugg som kan vara en risk i framtiden men behöver inte åtgärdas nu
	- Låg risk	Åtgärdas om tid finns

S a n n o l i k h e t	3	(4)	(7), (8)	(2), (3)
		(5)		(1)
			(6)	
	2			
	1			
		1	2	3
		Konsekvens		

## 10 Root Cause Analysis

Root cause analysis är en systematisk process för att identifiera grundorsaker till problem eller händelser och ett sätt att svara på dem. Denna metod använder man för att undvika att liknande problem uppstår senare.

- **Typ av test - exploratory testing**

Problem			Grundorsak			Lösningsförslag		
ID	Beskrivning	Kritiskt	Beskrivning	Sannolikhet	Information	Beskrivning	Risker	Var buggen hittades
P2-37	Felmeddelande poppar upp som säger: "Google Maps lästes inte in korrekt på sidan"	Allvarligt	Google maps saknar API nycklar	Allvarligt	När man använder inspektläget för att hitta orsaken till felet står det i consolen att API nyckel saknas	För att skaffa API nyckel behöver man koppla sitt google konto till Google Cloud Platform och den behöver verifiera att användaren inte är en robot genom att koppla sitt bankkonto till Google	Användaren får inte tillgång till Google kartor	Vi tycker buggen bör ha upptäckts i krav dokumentet eftersom det står i krav dokumenten att huset ska visas i Google maps
P2-59	Läs mer knappen öppnar pop up fönstret med information, istället så tas användaren till första sidan	Allvarligt	Läs mer knappens funktion fungerar ej.	Allvarligt	Klickar man på "Läs Mer" knappen under fliken "Priser" omdirigeras man till första sidan.	Omdirigerar användaren till fel plats	Användaren blir irriterad och letar efter andra alternativ	Vi tycker utvecklaren bör ha upptäckt buggen i utvecklingen av sidan. Eftersom det är en bugg som inte existerade i tidigare versioner av sidan
P2-43	Bilden på framsidan laddas inte upp	Irriterande	Failed to load resource: the server responded with a status of 404 (Not found)	Irriterande	Felmeddelande som kommer upp i inspekt läge - Failed to load resource: net::ERR_NAME_NOT_RESOLVED	Lösningen är att det ska lyckas att läsa in bilden på första sidan.	Att bilden inte dyker upp på framsidan	Utvecklaren bör ha upptäckt buggen. Eftersom det är en väldigt synlig bugg, första sidan är helt vit när man går in på hemsidan och vi tycker det är en bugg som inte är möjlig att missa
P2-38	Kontakt funktionen fungerar inte. Man kan inte kontakta personen som äger huset eller någon ansvarig	Allvarligt	Funktionen är inte färdigutvecklad	Allvarligt	Felmeddelande som kommer upp "There was an error. Please try again"	Användare har inte möjlighet att kontakta ägaren av hemsidan eller huset	Att man inte kan komma i kontakt med ägarna	Buggen bör upptäckas i utvecklingen av sidan. Eftersom vi tycker att det står i kravet att det ska finnas en kontaktformulär som ska skicka ditt meddelande till en specifik mejl

# Newton

P2-54	Kalendern skickar ingen bekräftelse mail efter man har bokat	Mindre allvarligt	Mejl funktion kan inte instansiera	Mindre allvarligt	Användaren ska få en bekräftelse via mejl	Som användare skall man få bekräftelsemail efter att man bokat	Användaren får ingen bekräftelse mejl och därmed osäker om bokningen gått igenom	Vi tycker detta är något som testare ska upptäcka eftersom det är vi som testar dessa funktioner
P2-56	Mindre information i engelska version, under fliken <b>"About Conil"</b>	Mindre allvarligt	Det är mindre information på engelska versionen under fliken "About Conil"	Mindre allvarligt	Utvecklaren har inte lagt till information på sidan	För att lösa detta så måste utvecklaren lägga till mer information som fattas på webbsidan (engelska versionen)	Användaren får inte tillräckligt med information	Det är utvecklaren som ska lägga till informationen på webbsidan tycker vi
P2-50	Det finns ingen kalender på <b>engelska versionen</b> , under fliken <b>"Prices"</b>	Allvarligt	Saknas en kalender i webbsidan (engelska versionen)	Allvarligt	Utvecklaren har inte implementerat kalendern i webbsidan	Kalenderns funktion måste implementeras till webbsidan	Användaren kan inte reservera lägenheten	Vi tycker utvecklaren ska upptäcka detta bugg eftersom det står i kraven vad webbsidan ska innehålla och vilka funktioner
P2-46	Stavfel under fliken <b>"Priser"</b>	Mindre allvarligt	Stavfel på webbsidan	Mindre allvarligt	Stavfelen är de två första rutorna, det står förnamn och efternamn	Det ska stå förnamn och efternamn istället för förnam och efternam	Användaren kan tolka sidan som icke professionell	Vi tycker detta är något utvecklaren ska upptäcka eftersom det är stavfel och ingen bugg som upptäcktes
P2-39	Otydlig text under fliken <b>"Priser"</b>	Irriterande	Texten är otydlig/ej synlig	Irriterande	Texten under "Högsäsong och lågsäsong" är otydlig/ej synlig	Texten ska vara synlig för användare	Användaren kan ge upp och leta efter andra alternativ	Vi tycker utvecklaren bör upptäcka detta bugg eftersom det är utvecklaren som har designat webbsidan

## 11 Tidsplan

Vi gjorde en tidsestimering på alla delar i projektet där vi gick igenom omfattningen av projektet i helhet.

- *Projektets tidsestimering från start-till-avslut.*

Testaktiviteter	Start	Avslut	Tidsuppfattning	Omfattning	Kommentar
Test Planering	18/3 -19	14/5 - 19	2 d	2 d	2 dagar för att slutföra dokumentationen av vår planering
Granskning av kravdokument	18/3 -19	14/5 - 19	2 d	1 h	Mindre än 1 timmes granskning av "krav".
Funktionstest – (Sprint 1-7)	20/4 - 19	14/5 - 19	5 d	2 h	Varje uppdatering av Wordpress kod tog oss 2 h att identifiera buggar på hemsidan.
Stress/prestandatest - (Sprint 1-7)	1/4 - 19	14/5 - 19	1 d	1 h	Det tog ca 1 h att lära sig om verktyget Jmeter. När man väl var igång med programmet kunde man få ut resultatet ganska snabbt.  Efter varje sprint testade vi prestandan på Conil och tog oss ungefär någon minut att få fram slutresultatet.
Skapande av Testfall	5/4 - 19	14/5 - 19	5 d	2 h	Vi skapade först en mall om hur testfallen skulle se ut sedan följde vi enligt mallen.
Enhetstest	22/4-19	1/5 - 19	2 d	4 d	Med hjälp av tutorials på nätet kunde vi implementera och exekvera scriptet.

## 12 Rekommendationer

Baserad på all testning vi har gjort rekommenderar vi inte att gå vidare till acceptanstestning då viktiga funktioner saknas. Dessa funktioner är affärskritiska, finns med som krav på kravdokumentet och är grunderna till webbsidan. I kontaktformuläret kan man inte skicka frågor till ägarna och kalendern som ska visa boendets tillgänglighet fungerar inte som den ska. Att denna funktion gör det den ska är viktigt för att undvika dubbelbokningar, effektivisera bokningssystemet och minimera missar i kommunikationen mellan kund och ägare.

-> [Kravdokumentet sektion 5.2.3 f](#)

-> [Kravdokumentet sektion 5.2.2 punkt 5](#)

Nedan två defekter är fortfarande öppna och anses som kritiska:

- Under fliken "PRISER" och rubriken "tillgänglighet & bokningsförfrågan" dyker ett varningsmeddelande upp efter att man har fyllt i sina uppgifter och skickat in dem.  
-> [Länk till bugg 51, 63](#)
- Under fliken "KONTAKT" och rubriken "KONTAKTA OSS" dyker ett felmeddelande upp efter att man har fyllt i sina uppgifter och klickar på knappen "SKICKA". -> [Länk till bugg 38](#)



## Bilagor

Länk till -  
[Teststrategi](#)

Länk till -  
[Testplan](#)

Länk till -  
[Veckologg](#)

Länk till -  
[Testfall](#)

Länk till -  
[Jira - rapporteringar](#)

Länk till -  
[Trello - tavla](#)

## Referenslista

Kurslitteratur -

[The art of software testing](#)