



UPPSALA  
UNIVERSITET

Självständigt arbete i informationsteknologi  
20 februari 2024

# Rapportmall för självständigt arbete

version vinter 2022

---

Sofia Cassel  
Björn Victor (bjorn.victor@it.uu.se)



UPPSALA  
UNIVERSITET

Institutionen för  
informationsteknologi

Besöksadress:  
Ångströmlaboratoriet  
Lägerhyddsvägen 1, hus 10

Postadress:  
Box 337  
751 05 Uppsala

Hemsida:  
<https://www.it.uu.se>

## Abstract

### Rapportmall för självständigt arbete

version vinter 2022

---

*Sofia Cassel*

*Björn Victor (bjorn.victor@it.uu.se)*

Abstract, always in English, about 10-20 lines. Do not use references, and do not use formulas if they can be avoided.

1. What is the problem/issue/subject?
2. How was the problem solved/attacked?
3. What are the results, how well was the problem solved?
4. How good are the results, how useful are they?

The abstract should be understandable without reading the whole report (and the rest of the report should be understandable without reading the abstract). You can reuse text/phrases from the Introduction.

Extern handledare: Hand Ledare, Firma Ment AB

Handledare: NN och Björn Victor

Examinator: Björn Victor

# Sammanfattning

Sammanfattning, alltid på svenska. Se till att det står samma saker i den svenska sammanfattningen och det engelska abstractet.

**Tips:** skriv först på det ena språket (dvs abstract *eller* sammanfattning), och när den versionen är klar, översätt den då till det andra språket. Lämna “det andra språket” tomt eller med en notering om att den skrivs när “det ena språket” är klart.

1. Vad är problemet, ämnet?
2. Hur angreps/löstes problemet?
3. Vad är resultaten, hur väl löstes problemet?
4. Hur bra blev resultaten, hur användbara är de?

Ca 10-20 rader. Använd inte referenser; ej heller formler om det går att undvika.

Sammanfattningen ska vara förståelig utan att läsa resten av rapporten, och resten av rapporten ska kunna läsas utan att läsa sammanfattningen. Det är helt OK att återanvända text från introduktionen.

# Innehåll

<b>1</b>	<b>Introduktion eller Inledning / Introduction</b>	<b>1</b>
<b>2</b>	<b>Bakgrund / Background</b>	<b>2</b>
<b>3</b>	<b>Background (!)</b>	<b>2</b>
3.1	Prototyping (!) . . . . .	2
3.2	Virtual Reality (!) . . . . .	4
3.3	Virtual Reality Prototyping (!) . . . . .	4
3.4	3D rendering tools (!) . . . . .	4
3.5	Unreal Engine (!) . . . . .	4
3.6	Unreal Engine in Automotive Industry (!) . . . . .	5
3.7	Einride (!) . . . . .	5
<b>4</b>	<b>Syfte, mål, och motivation / Purpose, aims, and motivation</b>	<b>5</b>
4.1	Avgränsningar / Delimitations . . . . .	7
<b>5</b>	<b>Relaterat arbete / Related work</b>	<b>7</b>
<b>6</b>	<b>Teori / Theory</b>	<b>8</b>
<b>7</b>	<b>Metod/Tillvägagångssätt/Tekniker eller Method/Approach/Techniques</b>	<b>9</b>
<b>8</b>	<b>Systemstruktur / System structure</b>	<b>10</b>
8.1	Tänk på följande . . . . .	10
<b>9</b>	<b>Krav och utvärderingsmetoder / Requirements and evaluation methods</b>	<b>10</b>
<b>10</b>	<b>DEL x: Implementation av XYZ</b>	<b>11</b>

<b>11 DEL x+1: Algoritm för igenkänning av stora fötter</b>	<b>12</b>
<b>12 DEL x+2: Användning av systemet</b>	<b>12</b>
<b>13 Utvärderingsresultat / Evaluation results</b>	<b>12</b>
<b>14 Resultat och diskussion / Results and discussion</b>	<b>13</b>
<b>15 Framtida arbete / Future work</b>	<b>13</b>
<b>16 Slutsatser / Conclusions</b>	<b>13</b>
<b>A Hur man gör appendix</b>	<b>14</b>
<b>B Några tips för LaTeX-användning</b>	<b>14</b>
B.1 BibTeX-tips . . . . .	16
<b>C Referenser</b>	<b>17</b>
<b>D Formler, figurer, tabeller, bilder, kod</b>	<b>19</b>
<b>E Språk, grammatik, stil</b>	<b>20</b>

# Hur ni använder detta malldokument

Titta i källdokumentet för diverse inställningar för författare, titel, etc. Läs också källdokumentet för instruktionerna i filerna `instr-X.tex` för olika värden på `X`.

*OBSERVERA* att de “fasta fält” som blir på svenska (trots att ni ställt in engelska med `babel`), som Examinator,Handledare, datum på framsidan osv, *ska* vara på svenska oavsett språk i rapporten. Abstract ska alltid vara på engelska, medan Sammanfattning alltid ska vara på svenska.

I flera appendix finns mer info och instruktioner som inte gäller själva rapportstrukturen.

För att slippa få med instruktionerna för rapportstrukturen i era inlämningar, ta bort `\input{instr-X}` för alla värden av `X` i källdokumentet.

**Tips:** ni kan använda separata filer för de olika delarna i er rapport på motsvarande sätt, men använd inte samma filnamn!

## Generellt

Varje numrerat avsnitt ska finnas med i er slutrapport, om inget annat anges. Välj rubrik på svenska eller engelska beroende på ert valda rapportspråk.

Om ni skriver på engelska ska titeln skrivas med första bokstaven i varje ord versal, utom “småord”. Exempel: *A Really Interesting Project on the Fundamentals of Shoes*<sup>1</sup>. (Detta gäller även titlar i referenser på engelska.) Rubrikerna i texten kan skrivas på detta sätt eller som på svenska (stor första bokstav i meningen), men *var konsekventa*.

Glöm inte att läsa kurslitteraturen [?, ?].

Undvik helst att lägga in en ordlista/glossary i början av rapporten. När man läser den kommer begreppen utan kontext och det kan vara svårt att förstå. Ni måste ändå förklara begreppen när de behöver förklaras (se appendix E, sid 22).

---

<sup>1</sup>Se t.ex. <https://en.wikipedia.org/wiki/Capitalization#Titles>

## Att göra

En onummerad sektion som beskriver läget för rapporten kan vara användbart i “veckans inlämning” för att underlätta feedbacken.

För att hantera “att-göra-listor” i rapporten kan LaTeX-paketet `todonotes` kanske vara användbart, både för att själva komma ihåg vad som ska ändras/läggas till, och för att visa det för läsarna under skrivandets gång. Se <http://ctan.org/pkg/todonotes> för mer info.

Naturligtvis är det generellt bra att också använda ett projekthanteringssystem som Trello<sup>2</sup> el.dyl. för att hålla reda på allt som är kvar att göra!

---

<sup>2</sup><https://trello.com/>

# 1 Introduktion eller Inledning / Introduction

Beskriv åtminstone samma saker som i abstract, men mer utförligt – typiskt 1-2 sidor. Spara tekniska detaljer till senare, eftersom läsaren inte är insatt än.

1. Vilket är området ni arbetar inom? Vad är problemet, ämnet, kontexten?
2. Varför är problemet viktigt/intressant att lösa?
3. Hur angreps/löstes problemet?
4. Vad är resultaten, hur väl löstes problemet?
5. Hur bra blev resultaten, hur användbara är de?

Tänk på att börja introduktionen med en mening eller ännu hellre ett helt stycke som “fångar” läsaren och motiverar läsaren att fortsätta läsa. *Vi har valt att göra ett projekt om X* är relevant för er, men kommer inte att vilja få någon att läsa vidare. Försök åtminstone få med någon slags bakgrund/kontext och (helst) motivation att fortsätta läsa. Typ *X är ett programspråk som tagit världen med storm. Vi vill utforska om man kan kombinera X med Y för att göra...*

Se till att ni *kommer till kritan snabbt* – man vill inte läsa igenom två stycken text innan man får veta vad ni tänker göra i ert projekt. Börja t.ex. *inte* med att presentera alla idéer ni inte valt – läsaren vill veta vad ni ska göra, inte vad ni inte ska göra. Använd gärna en bild som visar vad det är ni åstadkommit.

Översiktlig beskrivning av systemet och dess features ska vara under systemdesign / systemstruktur, inte i introduktionen.

Introduktionen bör vara begriplig för t.ex. en student i årskursen under, och gärna för en ännu bredare läsarkrets.

Avsluta gärna med en överblick över hela rapporten, där ni *kortfattat* beskriver vad de olika kapitlen handlar om.

**Tillkännagivande eller Tack / Acknowledgement** Tillkännagivande eller Tack (Acknowledgement på engelska) behövs inte alltid, utan bara ibland: har man fått avsevärd hjälp/feedback/stöd från någon (utöver det som kan väntas t.ex. av en handledare), har man fått speciellt tillstånd att använda något material eller utrustning, har man fått



ekonomisk sponsring (som t.ex. skulle kunna påverka den vetenskapliga opartiskheten). . . Avsnittet ska vara kort och kärnfullt, typiskt ett stycke. Finns det behov att skriva något längre, diskutera med handledaren först.

**Redovisning av arbetsfördelning / Declaration of division of labor** Här redovisar ni er arbetsfördelning inom gruppen, på en rimligt hög nivå men ändå någorlunda tydligt. *Exempel:*

Utvecklingsarbetet har fördelats jämnt i gruppen. NN har främst fokuserat på användargränssnittet, MM på databasoptimeringen, och NN har varit den mest drivande i utvecklingen av testmetodiken. Motsvarande delar av rapporten har framför allt skrivits av respektive person. I övriga delar har alla författarna varit engagerade och bidragit till design, utveckling, resultat, utvärdering och diskussion.

## 2 Bakgrund / Background

Här beskriver ni bakgrunden till ert projekt, d.v.s., det som leder fram till er problemformulering. Vilket är området, omgivningen, kontexten, bakgrunden för projektet (mer detaljerat och djupare än i introduktionen)? Beskriv området (t.ex. ljudbehandling, studieplaner, visualisering, autism...). Beskriv uppdragsgivare, om ni har (men inte för detaljerat). Tänk på att bakgrunden och problemet måste vara på en generell akademisk nivå och inte bara relaterat till en uppdragsgivare.

Tänk på att bakgrunden kan se längre tillbaka – hur löste man problemet förr? Innan man började datorisera? Ibland är det både viktigt och intressant (men ibland inte).

Efter att ha läst bakgrunden ska det vara lätt att förstå syfte/mål och att de är viktiga, och man ska förstå att ni är väl insatta i ämnet. Ett typiskt bakgrundsavsnitt är 1-3 sidor.

## 3 Background (!)

### 3.1 Prototyping (!)

In the manufacturing cycle, prototyping plays a crucial role. It assesses the fit, functionality, and form of the parts being prototyped. This phase is important because it provides a clear idea of the final product's appearance and allows for the identification of unne-

cessary elements in the design. By addressing these issues during the prototyping stage, it saves time and money that would otherwise be spent making corrections during the production phase.

Prototyping can be traced back to 1859, when a method known as photosculpture was invented by the French inventor Francois Willeme [?]. By surrounding an object with twenty-four cameras, each separated by 15 degrees, and taking photographs simultaneously, each photograph could be projected onto a screen. Using the images, a pantograph, and pieces of wood, 3D sculptures could be recreated from the photographs. This method of 3D prototyping enjoyed brief success but was eventually abandoned due to its labor-intensive nature.

Further on, in 1981, Hideo Kodama, who worked at the Nagoya Municipal Industrial Research Institute in Japan, released a paper describing what is likely the first 3D-printed object in history [?]. The paper outlines a system called “Automatic Method for Fabricating Cubic Shapes.” This method involved using a photo-hardening polymer to fabricate solid models by stacking layers 2 mm thick on top of each other, thereby creating the first 3D-printed object. Similarly, Charles Hull developed stereolithography in 1986 [?]. This method involves solidifying a liquid material when it is exposed to UV light, based on cross-sections of a 3D model. Unlike Hideo Kodama, Charles Hull patented his method and achieved greater academic and commercial success. However, Hideo Kodama was eventually awarded the Rank Prize, a prestigious award given to outstanding inventors, which he shared with Hull.

For prototyping, 3D printing is a widely used tool across various industries, including aviation, medical, automotive, etc. This technology is based on the concept of solidifying materials. Typically, the process begins with the creation of a 3D model in CAD or a similar system. These programs generate a file containing instructions for a rapid prototyping machine, usually a 3D printer, to create the 3D model in solid form [?]. There are several techniques for producing a prototype in the realm of 3D printing, including:

- Stereo lithography
- Selective Laser Sintering
- Fused Deposition Modeling

While these technologies differ in their specific processes, they share similar advantages, such as reducing product development and manufacturing costs, thereby increasing competitiveness [?]. Initially, these technologies and rapid prototyping were only accessible to large corporate firms, but today, 3D printers have become affordable for pri-

vate individuals as well. Before the widespread availability of 3D printing, prototyping required skilled engineers to work from 2D drawings.

### **3.2 Virtual Reality (!)**

Virtual reality is a tool that artificially simulates our senses. While the technology might seem new, the first VR technology was introduced by Ivan Sutherland in 1968 [?]. Today the VR industry is worth billions of dollars and is found in many industries.

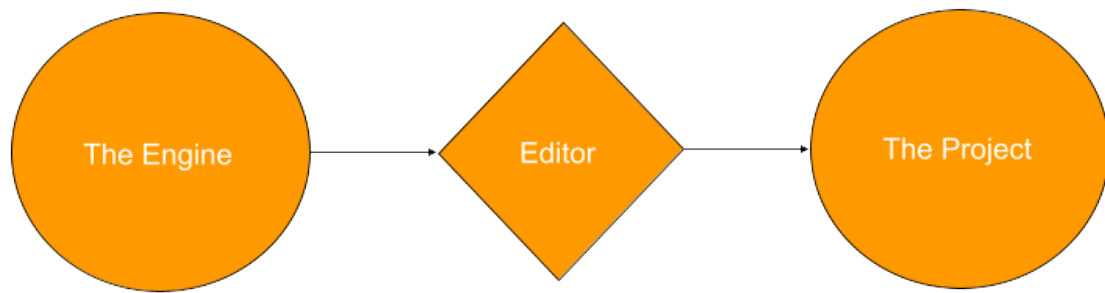
### **3.3 Virtual Reality Prototyping (!)**

### **3.4 3D rendering tools (!)**

3D rendering is used in multiple industries for various purposes, such as creating and animating special effects in movies, virtual prototyping, developing games, etc. Although the core technology behind 3D rendering is quite complex and requires advanced in-depth knowledge of coordinate systems, including the Cartesian coordinate system, and graphics along with complex algorithms, only a limited number of people possess this expertise. Consequently, software tools for simplifying the 3D rendering process have naturally emerged. Depending on the purpose, different tools have arisen: for 3D modeling, CAD has become a suitable option, while for 3D environments, Blender or Unreal Engine are more appropriate.

### **3.5 Unreal Engine (!)**

The development of 3D environments which is the main type of environment for VR can be done through multiple, though one tool that stood out was Unreal Engine. Originally created as a game engine, Unreal Engine is used as stated earlier in various industries. There exist multiple versions of Unreal Engine and as this is written the version 5.3.2 is the latest stable one. In Unreal Engine 5, the actual engine and project are separated to different folders with the editor as the separator but still part of the engine, a simple illustration of this is shown in figure 3.5.



---

**Figur 1** Figure 1: Engine structure

---

### 3.6 Unreal Engine in Automotive Industry (!)

Multiple car brands use Unreal Engine for different purposes. Volvo, a car brand with emphasis on safety, uses Unreal Engine to visualize and enhance sensor and safety. Apart from this, they also prototype human-to-machine interaction (HMI) functionality.

### 3.7 Einride (!)

## 4 Syfte, mål, och motivation / Purpose, aims, and motivation

Here we describe in principle your problem formulation. In this section, you should state the purpose, goals, and motivation for the project. These should not *int* be separate subheadings.

**Syfte.** What does the project aim to achieve? What is the overarching goal, the benefits, the effects of the project? (e.g., better health through control of food intake, simpler planning of studies...)

**Mål.** What should be delivered/implemented by the project, to achieve the purpose?

**Motivation.** Why is your project important? What is it important for, what external interests are there? How big is the problem, what are the consequences if it is not solved, how good would it be to solve it? What "gap" in the area does it fill? Why is your solution better/different than others?

Undvik att ta upp er interna motivation, som väldigt sällan är intressant för läsaren.

Se till att ni i detta avsnitt övertygar läsaren om att problemet finns, att det inte är löst, och att det är viktigt att lösa. Ju starkare argumentation och motivation (med källor) dess bättre.

- Visa att det finns ett problem.
- Visa att problemet är viktigt att lösa, att det behöver lösas.
- Visa att problemet inte redan är löst.

I det här avsnittet kan ni också börja beskriva etiska aspekter, hållbarhetsaspekter, och dataskyddsaspekter, men det finns förstås flera andra naturliga ställen att ta upp dem (*till exempel* sektionerna 7 och 8 (för att motivera olika val), 9 (för att motivera olika krav), eller 14 och 16 för att reflektera över resultatet, men kanske redan i sektion 1 eller 2).

Det är helt OK (och bra!) att också beskriva negativa/kritiska aspekter av ert projekt och arbete, inte bara positiva/goda.

Använd kursmaterialet för att få stöd att utveckla dessa aspekter. Läs t.ex. om forskningsetik [?] och dataskyddsprinciper<sup>3</sup>. Använd stödfrågor som t.ex. följande. **Se inte detta som ett frågeformulär**, men se till att hantera frågeställningarna – kanske inte just här, men på relevanta ställen i rapporten.

- Vilka är *direkta* intressenter och hur påverkas de? (användare, företag, kunder)
  - Vad krävs för att kunna använda er lösning? (kunskap, förmågor, resurser)
  - Vilka exkluderar ni?
  - Vad underlättar ni och vad gör ni svårare?
  - I vilka sammanhang kan er lösning användas och inte användas?
- Vilka är *indirekta* intressenter och hur påverkas de? (familj, samhälle, konkurrenter)
- Kan tekniken användas för “fel” syften?
- Om ni behandlar personuppgifter, varför behöver ni göra det, och för vilka ändamål? Har ni laglig grund, och följer ni GDPR-principerna?

---

<sup>3</sup>Integritetsskyddsmyndighetens webbsajt: <https://www.imy.se>

- Hur skyddar ni personuppgifter, hur minimerar ni mängden, och hur minimerar ni tiden de bevaras?
- Vilka resurser behöver systemet för att fungera? Vilka behövs för att underhålla systemet? Vad krävs för att avveckla systemet?
- Hur ser ett samhälle ut där er lösning används i stor skala?

När ni beskriver hållbarhetsaspekter av ert projekt, utgå från själva projektet och hur det hjälper olika typer av hållbarhet, och haka sen in referenser till FNs hållbarhetsmål om det är relevant. Att börja i FN-änden och sen beskriva hur ni bidrar till olika hållbarhetsmål ger ofta ett mer påklippt och artificiellt intryck.

## 4.1 Avgränsningar / Delimitations

Här beskriver ni vad ni *inte* gjort, alltså hur ni valt att begränsa er, och motiverar dessa avgränsningar. Detta förtydligar för läsaren som kanske hade förväntningar ni inte uppfyllt.

(I tidiga versioner, men *inte* i slutversionen, kan ni även beskriva vad som bara ska göras om tid/resurser/omständigheter räcker till. De sakerna kan ni då istället beskriva i Framtida arbete, sektion 15.)

## 5 Relaterat arbete / Related work

Här beskriver ni liknande system eller projekt, och förklarar hur de relaterar till ert. Alltså: vad vet ni om läget när det gäller “det större problemet” som projektet ska lösa? Vilka andra har försökt lösa liknande/närliggande problem, eller gjort relaterade/liknande saker/system? Referera! (Se Appendix C för mer om hur.)

Ha alltid ett första stycke som beskriver vilka typer av relaterade arbeten ni tar upp. Förklara/motivera varför *just dessa* beskrivs och hur ni valt ut dem, så att läsaren övertygas om att ni gjort ett vettigt urval av relaterat arbete (och inte bara beskriver de första google-träffarna).

Liksom för bakgrunden kan relaterat arbete också gå längre tillbaka. Det är inte nödvändigtvis bara datorbaserade/appbaserade/etc lösningar som är relaterade.

Några olika typer av relaterat arbete:

- Direkt relaterade till ert system (t.ex. alternativ till ert system) – och hur är de relaterade?
- Relaterade till era metoder (t.ex. olika sätt att göra positionering)
- Relaterade till syftet/grundproblemet (vad ska man använda t.ex. positioneringen till?)

Tänk också på:

- Relaterat arbete bör vara på en generell (gärna akademisk) nivå och inte bara relaterat till en uppdragsgivare, en programmeringsplattform, eller ett särskilt sätt att angripa problemet.
- När ni jämför ert system med andra, se till att läsaren fått en översikt över vad ert system är först (t.ex. i inledningen) så att vederbörande kan göra en kvalificerad bedömning.
- Beskriv vad varje relaterat arbete är (t.ex. en app, en undersökning...), vad deras resultat var, **och hur det relaterar till ert arbete.**

(Ovanstående är ungefär max-storlek på saker i en punktlista – är det mer text är det oftast bättre med riktiga paragrafer.)

Ofta är det bra att gruppera relaterade arbeten (t.ex. appar som löser liknande problem, eller andra angreppssätt än tekniska). Ofta är det effektivt att efter en grupp relaterade arbeten summera hur de relaterar till ert (t.ex. “dessa appar har dessa liknande finesser, men ingen av dem hanterar X som är en av våra huvudpoäng”).

Efter att ha läst det här avsnittet ska läsaren förstå att ni satt er in i tidigare arbete, och t.ex. inte uppfunnit hjulet på nytt.

## 6 Teori / Theory

Om ert projekt berör eller använder teoretiska ramverk el.dyl. som en student i årskurs 2 inte rimligen har god koll på kan ni ha med ett avsnitt om dem. Använder ni inga speciella teorier så kan ni skippa det här avsnittet.

Använd alltid en mer beskrivande rubrik.



UPPSALA  
UNIVERSITET



UPPSALA  
UNIVERSITET

---

**Figur 2** Till vänster syns universitetets logotyp i färg, till höger i svartvitt.

---

## 7 Metod/Tillvägagångssätt/Tekniker eller Method/Approach/Techniques

Här beskriver ni vilka metoder/verktyg/tekniker/approacher ni använt för att lösa problemet / besvara frågeställningen. Vilka metoder har ni konkret använt för att lösa problemet/bygga systemet? Vilka tekniker/verktyg använde ni?

Observera att det inte är samma sak som att beskriva *hur* ni använde teknikerna/verktygen: det kommer i Del X, implementationsdelen (se avsnitt 10).

Glöm inte att *motivera* era val av metoder. Finns det flera rimliga alternativ? Beskriv varför ni inte valt dem (t.ex. varför er valda metod är bättre). Visa att det är rimligt att använda just detta tillvägagångssätt. Det gäller även i det fall det är givet på förhand vilken teknik ni ska använda (t.ex. vilket programspråk) för att det ska passa i ett sammanhang eller existerande system (t.ex. ett som ny bygger vidare på). Även om man har en given teknik kan man alltså behöva förklara att en annan egentligen vore bättre – men att omständigheter gör att ni ändå måste välja den givna tekniken.

Det är ofta bra att börja med att förklara vad ni valt för teknik/verktyg, och därefter motivation och alternativ. Om man börjar med alla alternativ och väntar med att förklara vad man valt till sist, blir det inte lika enkelt att läsa.

Detta avsnitt ska *inte* innehålla information om hur gruppen organiserat arbetet (github, trello, jira...) *om* det inte är relevant för resultatet (och det är det oftast inte).

Använd tydliga underrubriker, t.ex. “Ramverk för webbplatsen” snarare än “Wordpress”, eller “Databashanterare” snarare än “MongoDB”.



## 8 Systemstruktur / System structure

Beskriv strukturen både internt (hur ert eget system är uppbyggt) och externt (vilka andra system ert system kommunicerar med). **Använd figurer** (och text)!

- Vilka delar består systemet av? (T.ex. databas, webbinterface, AI-modul, grafik...) Vilka kommunicerar med vilka, beror av vilka, innehåller vilka andra?
- Vilka delar fanns färdiga att använda/anpassa, vilka utvecklade ni själva? Visa tydligt, gärna grafiskt.
- Finns olika alternativa byggblock eller designval? Vilka är argumenten för/emot valen?
- Hur kommunicerar delarna, vilka protokoll och/eller dataformat används? (Beskriv mer detaljerat i senare, i Huvuddelen.)
- Finns det olika typer av användare/motsv? (T.ex. administratörer resp slutanvändare?)

### 8.1 Tänk på följande

Var inte för tekniskt detaljerade här. Tanken är att ge en översikt över systemet. Ni behöver inte beskriva objektmetoder etc. i detalj (om de inte är nya och avgörande för resultatet). Tekniska detaljer och implementation beskriver ni snarare i Huvuddelen.

Se till att ni använder *samma terminologi* i figurer som visar systemet som i texten.

*Anknyt figurerna till texten* på ett tydligt sätt. Om ni t.ex. har separata underrubriker som beskriver olika delar/aspekter av systemstrukturen med tillhörande figur, välj antingen en underrubrik per del i figuren eller använd helt andra underrubriker. Annars kommer läsaren att undra var underrubriken som beskriver del X är, när det finns underrubriker för alla andra delar.

## 9 Krav och utvärderingsmetoder / Requirements and evaluation methods

För de olika funktionaliteterna (och/eller motsv) i ert system, hur ska ni avgöra om de är tillräckligt bra utförda/implementerade? Var går gränsen för "tillräckligt bra"? (Eller

när är de “för dåliga”?)

Koppla kraven till syfte/mål, eller beskriv varför det inte går/är rimligt att utvärdera.

Skilj på mål och krav. Beskriv här vilka krav ni har satt upp för ert system, och hur ni ska utvärdera om de är uppfyllda. Beskriv *inte* funktionalitet hos systemet eller mål/syfte: de beskrivs ju i andra delar av rapporten.

*OBS:* Skilj på funktionalitet (vad ska systemet kunna göra) och krav (hur bra ska systemet vara). Själva funktionaliteterna har ni redan beskrivit i systemstrukturen eller huvuddelen nedan. (Har ni krav på saker ni beskriver först i huvuddelen kan ni lägga det här avsnittet efter huvuddelen.)

Att en funktionalitet överhuvud taget finns blir ett “binärt” krav som ni inte behöver utvärdera – ni vet att ni gjort det eller inte och beskriver helt enkelt det i ett annat kapitel – men vilken funktionalitet har ni krav på *hur bra* den ska vara (med olika tolkningar av “bra”)? De kraven ska beskrivas här.

Skriv tydliga krav *som går att utvärdera*. (Hur snabbt? Hur många användare? Hur strömsnålt? eller vad som är relevant). Se till att kraven är väldefinierade (vad betyder “snabbt” eller “effektivt”?)

Beskriv hur utvärderingen ska gå till (automatiserade belastningstester, mätningar, enkäter, fokusgrupper. . .). Beskriv hur externa intressenter involveras i utvärderingen.

Om ni upptäcker att det är svårt att utvärdera något kanske det beror på att det inte är ett väldefinierat krav. Att t.ex. säga att “systemet ska vara flexibelt och välskrivet” är svårt att kolla, även om ni kanske provar med kodgranskning. Om ni formulerar det tydligare är det mindre svårt: vilka aspekter var det ni tittar på när ni gör kodgranskningen, har ni krav på de aspekterna? Beskriv isåfall dem, och beskriv kodgranskning som en metod (gärna med referens). Att systemet är “modulärt och väldokumenterat” kan förtydligas till att ni följer en viss kodstandard (beskriv då vilken, med referens).

Krav och utvärdering kan beskrivas tillsammans (räkna upp krav och hur det utvärderas) eller separat (först alla krav, sen hur de utvärderas). Det är viktigt att det är tydligt, och att ni inte har krav utan någon utvärderingsmetod.

## 10 DEL x: Implementation av XYZ

Mellan introduktion och avslutning finns ett eller sannolikt *flera* avsnitt (“huvuddelen”) som innehåller själva bidraget eller implementationen. Ni får själva välja passande rubri-

ker (INTE “Huvuddel” eller “Bidrag”). Rubrikerna i huvuddelen ska tillsammans med titeln ge en idé om vad som berättas, en “berättelse”. (Exempel: “Algoritm för automatisk igenkänning av stora fötter”, “Design av databasen för användardata”, “Optimering av minnesanvändning”, “Implementation av djupinlärningssystemet” etc.)

Här kan ni beskriva implementationen, hur systemet används, etc. Använd väldigt gärna figurer, skärmbilder, tabeller och andra illustrationer!

Det kan vara användbart att koppla strukturen på denna del till systemstrukturen och dess olika delar.

Beskriv t.ex. centrala datastrukturer och hur de används. Beskriv hur de olika metoderna (från avsnitt 7) kommer till användning.

Beskriv gärna felhantering och riskanalys: vad kan gå fel när systemet kör/används, vad kan bli följden, och hur hanteras detta?

## **11 DEL x+1: Algoritm för igenkänning av stora fötter**

Se avsnitt 10.

## **12 DEL x+2: Användning av systemet**

Se avsnitt 10.

...

## **13 Utvärderingsresultat / Evaluation results**

Beskriv resultaten av utvärderingen, när ni tillämpar de utvärderingsmetoder ni beskrivit i avsnitt 9, och relatera utvärderingsresultaten till kraven i samma avsnitt.

Skilj på utvärderingsmetoder och utvärderingsresultat, och mellan utvärderingsresultat (dvs resultat av användningen av utvärderingsmetoderna) och det större resultatet (nedan).

## 14 Resultat och diskussion / Results and discussion

Här beskriver ni först era resultat, vad ni åstadkommit. Hur bra blev det? Sedan granskar ni era resultat kritiskt. Varför blev det som det blev? Var resultaten rimliga/bra/dåliga/oväntade...? Vad hade man kunnat göra annorlunda? Hur relaterar era resultat till liknande arbeten?

- Visa att utvärderingen är rimlig.
- Visa att utvärderingen, resultatet och analysen är vetenskapliga och ingenjörsmässiga.

Relatera till mål och syften etc i avsnitt 4, använd gärna eventuell teori i avsnitt 6 för att förklara.

I det här avsnittet kan ni inte lyfta fram nya resultat, fakta el.dyl, bara beskriva och diskutera det ni beskrivit tidigare.

## 15 Framtida arbete / Future work

Här beskriver ni potentiella framtida utvecklingar av systemet. Var finns förbättringspotential och vad kan man bygga vidare på? Vilka intressanta utvidgningar hann ni inte med?

Observera att tidsplanering, relation till kursmål etc. *inte* hör hemma i slutrapporten.

## 16 Slutsatser / Conclusions

Här sammanfattar ni och upprepar ert bidrag (resultaten av ert projekt) och förklarar dess vikt och användning. Vad var viktigt/nytt/intressant? (INTE i termer av vad ni lärde er, utan för den som läser rapporten, funderar på att göra ett liknande system, vidareutveckla ert system, etc.)

Summera de viktigaste delarna av projektet, men inte copy-paste utan gärna med en ny formulering. Sätt gärna de viktigaste bitarna i ett större perspektiv eller sammanhang.

## A Hur man gör appendix

Appendixar kan vara bra för bilagor som enkätundersökningar, större kodavsnitt, etc.

Appendix läggs efter referenslistan, och ska börja på en ny sida. Använd `\newpage` för att göra ett sidbrott där resten av nuvarande sida är tom. Skriv sen `\appendix` för att markera att resten är appendix, och använd sen vanliga `\section{}` för varje appendix, som kommer att “numreras” A, B, C osv.

För att referera till ett appendix, gör likadant som till ett avsnitt (se instruktion på följande sida), till exempel: “se appendix B för tips om LaTeX-användning (see Appendix B for tips about LaTeX usage).”

## B Några tips för LaTeX-användning

Ett enkelt sätt att använda/installera LaTeX för MacOS är TexShop<sup>4</sup>.

För **samarbete** i delade dokument när man skriver LaTeX-dokument använder somliga Overleaf (<https://www.overleaf.com/>, tips på liknande system är välkomna), men det funkar också att använda git och vanliga texteditorer (t.ex Emacs). Det är oftast smart att dela upp dokumentet i flera (t.ex. ett per kapitel) som sen inkluderas med `\input{kapitel}`. (Se rapportmallen för ett exempel på detta.) Ett tips är också att slå på radbrytning i texteditorn, så att konflikter vid incheckningar hanteras per kort rad istället för per jättelång rad.

**Läs också i Wikibooks** (<http://en.wikibooks.org/wiki/LaTeX>), **missa inte** Appendix om “Sample LaTeX documents” (men använd alltid rapportmallen som bas).

**Citat-tecken** skriver man med ```foo''` (dvs två bakåtnuttar före, och två vanliga fnuttar efter). LaTeX gör så att det blir snyggt: “foo”.

När man skriver på svenska behöver man ibland “visa” var ord (speciellt såna med med åäö) kan **avstavas** genom att använda `\-` (liknande *soft hyphen*): ämnesöversiktsintroduktion avstavas med några sådana instuckna på rätt ställen istället som ämnesöversiktsintroduktion

ämnes\-över\-sikts\-intro\-duk\-tion

---

<sup>4</sup><http://pages.uoregon.edu/koch/texshop>

För att slippa göra detta varje gång man skriver samma ord, kan man använda kommandot `\hyphenation`. Skriv då, före `\begin{document}` dvs i den s.k. *preamblen*, t.ex.

```
\hyphenate{ämnes-över-sikts-intro-duk-tion}
```

dvs med vanligt bindestreck på alla ställen ordet kan stavas av, så kommer alla förekomster av ordet att avstavas enligt mönstret.

För att formatera **URLer** bättre (så att t.ex. radbrytning blir snyggare), skriv t.ex. `\url{https://www.it.uu.se/research/group/concurrency}` i texten eller referensen.

För att **referera** till avsnitt, figurer, tabeller etc, använd `\label{markör}` för att “sätta ett märke” i text eller figur, och `\ref{markör}` för att referera till den (se exempel nedan). Ibland kan man behöva underlätta för läsaren att hitta det man refererar till, och då är macroet `\vref` väldigt kraftfullt.<sup>5</sup> Läs mer på sidan 19 om hur man benämner avsnitt, figurer osv.

```
\section{Motivation}
\label{sec:motivation}
```

följt av

Som vi nämnt i avsnitt~\ref{sec:motivation}...

eller på engelska (notera “Section” med stor inledande bokstav)

As already mentioned in Section~\ref{sec:motivation}...

Se också appendix A på föregående sida för hur man gör med appendix.

För att få referenser att inte hamna först efter ett **radbrott**, använd icke-brytande space såhär~\cite{fin-bok}, där tilde-tecknet ~ alltså gör ett obrytbart space. Detta är i princip också alltid rätt att använda före siffror (och i stora tal på engelska, t.ex. 100~000 för 100 000), och förstås också före \ref{fig}.

Använd *aldrig* dubbel-backslash `\\` för att få avbrott mellan stycken. Använd alltid dubbel ny rad för detta. Använd bara `\\` i tabeller o.dyl.

---

<sup>5</sup>Se <https://www.ctan.org/pkg/varioref> för mer info om varioref.

För att göra ett **sidbrott** där resten av sidan blir tom, använd `\newpage`, använd inte `\pagebreak`. Det senare är till för att finjustera var latex gör ett automatiskt sidbrott, inte för att avsluta en halvfull sida.

Använd aldrig *math mode* (dvs `$Text$`) för att få kursiv text eller för flerbokstavsvariabler i matematiska uttryck, eftersom text i math mode tolkas som en multiplikation av de olika bokstäverna och då får konstiga mellanrum – det blir särskilt tydligt där LaTeX normalt skulle ha gjort ligaturer (som *fi*). Använd istället `\textit{Text}` (*Text*) eller `\textsl{Text}` (*Text*), eller kanske `\textrm{Text}` (*Text*, speciellt i matematiska uttryck) eller `\textsf{Text}` (**Text**, speciellt för kod).

## B.1 BibTeX-tips

För att hantera bibliografi (**referenser**) på ett smidigt sätt, använd BibTeX! Läs mer i [http://en.wikibooks.org/wiki/LaTeX/Bibliography\\_Management#BibTeX](http://en.wikibooks.org/wiki/LaTeX/Bibliography_Management#BibTeX), nedan om referenser, och `IEEEtran_bst_HOWTO.pdf` i `directory IEEEtran/bibtex` i detta Github-repository.

För att se till att BibTeX inte gör namn, förkortningar etc till lowercase, använd `{ }` och skriv typ:

```
title = {The {DSP} of {N}ewton applied to {iOS}}
```

Skriv alltid månader för publikation med de inbyggda förkortningarna, typ:

```
month = jun
```

istället för `{jun}` eller `"jun"` eller `"June"` eller `"Juni"`. Då kan nämligen bibliographystyle styra hur det förkortas etc.

Kom ihåg att separera författarnamn med “and”, inte med komma. Ibland behöver BibTeX extra hjälp att förstå vad som är för- och efternamn, och *då* är komma rätt att använda. Exempel (se [?] i referenslistan):

```
{Whitehead, Alfred North and Bertrand Russel}
```

Ett verktyg för att hantera BibTeX-filer i MacOS är BibDesk<sup>6</sup>.

---

<sup>6</sup><http://bibdesk.sourceforge.net/>

## C Referenser

Se också kap 8.5 i Dawson [?].

**OBS: viktigt!** Det finns åtminstone tre syften med utformningen av referenserna och referenslistan.

1. Man ska hitta referensen (från texten) i referenslistan.
2. Man ska förstå vad som refereras (vilken typ av referens det är) så att man kan värdera den.
3. Man ska kunna hitta referensen i verkligheten.

Eftersträva alltid att uppnå alla tre.

Använd numeriska referenser (IEEE-stil [42]) eller nyckelordsbaserad [Lam86], inte fotnotstil. Referenserna sorteras alfabetiskt efter författare/motsv i referenslistan. I LaTeX, använd `\bibliographystyle{IEEEtranS}` eller `{IEEEtranSA}` (eller liknande), se rapportmallen. **Börja** med att använda `{IEEEtranSA}` som tydligare visar när viss info saknas i bibtex-entries (t.ex. år och författare).

För att göra inställningar för `\bibliographystyle{IEEEtranS/SA}` kan man använda ett speciellt bibtex-entry som heter `@IEEEtranBSTCTL`, se mer info i filen `IEEEtran_bst_HOWTO.pdf` i directoryt `IEEEtran/bibtex`, avsnitt VII, eller sista biten av `IEEEexample.bib` i samma directory.

Referenserna skrivs i direkt anknytning till det som föranleder referensen (t.ex. ett påstående eller resultat), före eventuellt skiljetecken, och med ett fast mellanslag till föregående ord. I LaTeX, skriv `\cite{lam86}` för att få en “non-breaking space”. Se också rapportmallen, och sista stycket på sid 211 i Dawson [?].

Man ska alltså *inte* skriva referenserna efter ett längre stycke (som vissa verkar lära sig att göra, någonstans). Det gör det oftast otydligt vad som egentligen är hämtat från, eller styrks, av referenserna. I vissa fall kan man vilka göra en kort sammanfattning av vad en författare skriver i en artikel el.dyl., men att bara lägga på en referens sist i stycket är inte tillräckligt tydligt. Det är mycket bättre och tydligare att inleda stycket med att skriva något i stil med “Lisa Lagom beskriver~\cite{lagom-bok} hur X beror av Y och i sin analys visar hon i detalj hur sambandet ser ut...”.

Att upprepa samma referens ofta i ett stycke (kanske efter varje mening) gör det mer svårläst. Försök att skriva om stycket så att det blir tydligt att det hela bygger på referen-



sen, som lämpligen används tidigt. Exempel: “I en undersökning av WHO~\ref{who} beskrivs följderna av XYZ och de indirekta risker som uppkommer”, och därefter kan de olika följderna och riskerna tas upp i samma stycke utan att det blir otydligt.

När man refererar till “tjocka” saker som böcker är det lämpligt att ange sidnummer (som \cite[sid 211-214]{dawson:projects-in-computing}) som blir [?, sid 211-214], eller kapitel (som \cite[kapitel 5]{...}), men för “tunnare” saker behöver man bara göra det för att speciellt peka ut om man t.ex. menar en viss del av referensen (kanske den tar upp tre olika sätt att göra X och man vill peka på det 3:e, inte de första två).

Upprepa inte samma referens (bok, artikel etc) i referenslistan för olika avsnitt, kapitel etc i referensen, utan gör på det sätt som beskrivs i stycket ovan.

Man skriver normalt inte ut titeln på det man refererar om det inte är ett speciellt viktigt verk, typ *Principia Mathematica* [?] eller *On Computable Numbers* [?]. Skriv alltså inte “I artikeln *How to increase efficiency in databases* [Ref21] beskriver författarna hur man ökar effektiviteten i databaser...” utan snarare “Reffersen beskriver [Ref21] olika sätt att öka effektiviteten i databaser...”.

För mer info om vilken info som behövs för olika typer av referenser, se avsnitt 8.5.3 i Dawson [?, ?]. (För att referera till flera saker samtidigt (som nyss) skriver man flera BibTeX-nycklar i samma \cite.) Där beskrivs också [?, sid 230] hur man refererar till vad någon bara har sagt, utan att det finns publicerat (“personal communication”). Den typen av referenser är förstås svåra att kolla upp för läsaren, och typiskt ganska svaga, så det är viktigt att det framgår hur “prominent” personen som sagt saker är, hur tungt dennas ord ska vägas.

Observera att nyhetsartiklar (tidningsartiklar och motsvarande) nästan alltid har ett publiceringsdatum, som ska visas (t.ex. i note-fältet), och oftast också en artikelförfattare.

När ni använder referenser som finns i DiVA, utgå från de BibTeX-data ni får genom att klicka “Exportera” ovanför titeln och välj BibTex, men *dubbelkolla* att resultatet blir bra! Specifikt för tidigare Självständiga arbeten i IT: ersätt bibtex-typen @misc med @techreport och fältet series med type. Lägg gärna till själva diva-länken (“Permanent länk” under “Länk till poster”) som url-fält. Se exempel [?, ?] i refs.bib.

Använd inte URLer till universitetets ezproxy-tjänst (t.ex. www-sis-se.ezproxy.its.uu.se), eftersom den bara kan användas av personer med UU-konto. Använd en direkt länk eller en DOI-länk istället.

Använd inte direktcitat, såvida inte den exakta formuleringen är viktig. Skriv hellre ett

referat av vad någon sagt. (Se Dawson [?, ?].)

Om referenslistan huvudsakligen innehåller referenser till “mer info” av typen `www.wordpress.org`, `www.w3c.org`, `developer.android.com`... men få referenser som stöder resonemang, motivation, argument etc (jfr Workshoparna), är det antagligen ett tecken på att det finns få resonemang, motiveringar och argument som behöver stödjas. Då behöver man med största sannolikhet resonera, motivera och argumentera mera!

Även om en referens har en URL till själva texten är det inte nödvändigtvis en webbreferens, utan ibland en artikel/bok el.dyl som råkar vara tillgänglig på nätet. Den ska då beskrivas som artikel/bok/el.dyl (men förstås gärna med URLen) så att man kan göra en preliminär värdering av referensen redan när man läser referenslistan.

Använd gärna Wikipedia eller en populärvetenskaplig artikel som en *start* för att läsa om något, men ansträng er att hitta stabilare referenser när ni skriver om detta något. Kolla vilka referenser artikeln använder, läs och använd (åtminstone någon av) dem istället.

Titta på er referenslista och reflektera över vilka typer av referenser ni använt. Är det bara onlinereferenser? Är det bara “läs-mera-här”-referenser? Eller har ni använt referenser på ett bra sätt för att underbygga era val, resonemang, argument och påståenden?

Försök hitta författare och publiceringsdatum (år, månad) även för webbreferenser, och ange **alltid** när de lästes, eftersom de kan uppdateras när som helst. Ett exempel är [?] (se `refs.bib`).

## D Formler, figurer, tabeller, bilder, kod

Formler, figurer, tabeller, bilder och ekvationer måste beskrivas. Det betyder t.ex. att varje symbol måste vara förklarad i texten.

Låt figurer och tabeller (“floats”) hamna där LaTeX tycker att de ska, och justera bara placeringen i slutversionen och om det verkligen behövs, t.ex. för att de flyter iväg flera sidor.

Figurtexter (captions) ska beskriva vad vi ser i figuren, inte bara vad det är för slags figur. Att skriva “Systemstruktur” eller “The structure of our system” för en bild av systemstrukturen är inte tillräckligt. Hjälp läsaren förstå genom att också (eller istället) beskriva innehållet, t.ex. “De gröna cirkelarna representerar användare, och komponenter med skuggad bakgrund är externa. Indata kommer från vänster, och utdata levereras till höger”. Det räcker alltså *inte* att beskriva figuren i löptexten – men naturligtvis ska den beskrivas där också.

I engelsk text skriver man “Figure 3”, inte “figure 3”, eftersom det fungerar som ett namn på figuren (och motsvarande för Table, Section, Appendix osv).

Alla figurer och bilder som inte är era egna måste ha referenser till källan.

Rapportmallen är inställd så att figurer presenteras med en linje över, en under, och en mellan figurtexten och själva bilden. För andra presentationer, se `\floatstyle` (googla) – exempelvis ger `\floatstyle{boxed}` istället en ram runt figuren.

Om ni inkluderar kodsntuttar, se till att de är relevanta och kommenterade, så att man förstår. Alternativt, för korta snuttar: ge motsvarande förklaring i texten. Använd vettigt latex-bibliotek för kod, t.ex. `listings`.

## E Språk, grammatik, stil

### Pronomen

- Det är OK att skriva “Vi”! Det blir ofta mer svårläst om texten skrivs i passiv form istället.
- **Inte alla läsare är män.** Skriv därför inte “han”, “hans”, “denne” etc. Använd könsneutrala pronomen eller ord som “vederbörande”, “användaren” etc.
- På engelska, undvik det informella *you* som översättning av “man” – undvik också *one* som kan upplevas som alltför formellt (drottningen uttrycker sig så). Skriv ut vem som menas, t.ex. “the user” el.dyl., eller använd “they” (även i singular).

### Ordval

- **Undvik talspråk** “så”, “två stycken saker”, “ifrån”, “utav”, “vart”, “kommer göra/vara” (istället för “kommer att göra/vara”, . . . **Kolla på Wikipedia-sidan** “Vanliga språkfel” [?].
- Undvik värderande uttryck som enkelt, uppenbart.

### Grammatik

- Undvik att sluta meningar med en preposition (t.ex. *med* eller *with*).

- Semikolon är **inte** en variant av kolon eller komma; semikolon kan endast användas där ni normalt sett skulle använt punkt, men vill fortsätta på samma mening. För att undvika problem, **undvik semikolon helt**.
- Efter uttryck som “för det första...”, “one alternative is...” måste följa “för det andra...” “another alternative” (inte “slutligen”, “dels”, “another option” eller något annat). Tänk också på “firstly ... secondly” resp. “first ... second”, inte “first ... secondly” eller något annat.
- Om ni skriver på engelska kan verktyget Grammarly ([www.grammarly.com](http://www.grammarly.com)) vara användbart för att kolla grammatik, för svensk text kan man prova att klippa-och-klistra texten till ett dokument i Word, som har inbyggd koll av grammatik och stavning.
- På engelska, se upp med skillnaden mellan plural-s och genitiv-s, hur man använder dem tillsammans, och hur det blir när grundordet slutar på s.
- På engelska måste ni använda korrekta verbformer beroende på om subjektet är en eller flera saker (“it has” men “they have”).

## Syftningar

- Skriv inte meningar som börjar med “Detta på grund av” eller “Detta eftersom...” – det blir ofta inte fullständiga meningar och det är ofta inte klart vad “detta” syftar på.
- Var försiktig med uttryck som “this approach”, “detta system”, etc. och kontrollera att det är uppenbart vad detta/this refererar till. Be någon icke-gruppmedlem läsa och kolla!

## Tempus

- Använd inte framtid (futurum); skriv rapporten i nu- eller dåtid och var konsekventa (Vi gör... eller Vi har gjort..., inte Vi kommer att göra...).
- Var noggranna med valet av dåtid eller nutid: skriv inte i nutid om saker som inte nödvändigtvis gäller när texten läses (t.ex. om fem år), om det inte är tydligt *när* det gäller. Gör det tydligt när det gäller.

## Definitioner

- **Förklara begrepp innan ni använder dem**, hänvisa inte *bara* läsaren till ett senare avsnitt (men ni kan naturligtvis också hänvisa till mer detaljerade förklaringar som kommer senare i texten). Första gången ett begrepp nämns måste alltså åtminstone en kort förklaring finnas.
- När ni introducerar nya koncept (sådant ni inte har diskuterat tidigare), gör inte det “i förbifarten”, utan se till att ni **förklarar ordentligt**. Alltså: “Vi använder X (ett häftigt nytt programmeringsspråk) för att göra Y” fungerar inte. Beskriv först konceptet ni använder, och använd det sedan. Typ “X är ett viktigt nytt programmeringsspråk. Vi använder X för att göra Y.”
- Använd inte olika synonymer för det ni har utvecklat (tjänsten/projektet/systemet), utan bestäm er för vad ni kallar det ni har gjort.
- Det kan vara bra att kursivera nya begrepp första gången de används, men normalt bör man inte kursivera *alla* förekomster.
- Undvik helst att lägga in en ordlista/glossary i början av rapporten. När man läser den kommer begreppen utan kontext och det kan vara svårt att förstå. Ni måste ändå förklara begreppen när de behöver förklaras (se punkten ovan).

## Förkortningar och kontraktioner

- **Var konsekventa** med hur ni skriver förkortningar och begrepp (c++ eller C++, android och Android t.ex.) Tumregel: namn skrivs med inledande stor bokstav (Android, inte android), förkortningar med stora bokstäver (XML, inte Xml).
- Undvik kontraktioner (sammandragningar) på engelska: skriv “do not” istället för “don’t”, och “is not” istället för “isn’t” osv. Var *extra försiktig* med “it’s” och “its” och “it is”, och välj rätt.

## Till sist

Undvik att skriva så här långa punktlistor, och punktlistor med så här mycket text.