



UPPSALA
UNIVERSITET

Självständigt arbete i informationsteknologi
8 april 2021

Rapportmall för självständigt arbete

version VT 2021

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



UPPSALA
UNIVERSITET

Institutionen för
informationsteknologi

Besöksadress:
ITC, Polacksbacken
Lägerhyddsvägen 2

Postadress:
Box 337
751 05 Uppsala

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

Abstract

Rapportmall för självständigt arbete

version VT 2021

Sofia Cassel

Björn Victor (bjorn.victor@it.uu.se)

Abstract, always in English, about 10-20 lines. Do not use references; 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.

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

1	Introduktion eller Inledning / Introduction	1
2	Bakgrund / Background	2
3	Syfte, mål, och motivation / Purpose, aims, and motivation	2
3.1	Avgränsningar / Delimitations	3
4	Relaterat arbete / Related work	4
5	Metod/Tillvägagångssätt/Tekniker eller Method/Approach/Techniques	5
6	Systemstruktur / System structure	5
6.1	Tänk på följande	6
7	Krav och utvärderingsmetoder / Requirements and evaluation methods	6
8	DEL x: Implementation av XYZ	7
9	DEL x+1: Algoritm för igenkänning av stora fötter	7
10	DEL x+2: Optimering av minnesanvändning	7
11	Utvärderingsresultat / Evaluation results	7
12	Resultat och diskussion / Results and discussion	8
13	Slutsatser / Conclusions	8
14	Framtida arbete / Future work	8

A	Hur man gör appendix	10
B	Några tips för LaTeX-användning	10
B.1	BibTeX-tips	11
C	Referenser	12
D	Formler, figurer, bilder, kod	14
E	Språk, grammatik, stil	14

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 som inte gäller 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*¹. (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 [Daw15, Daw09].

Att göra

En 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. Se <http://ctan.org/pkg/todonotes> för mer info.

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

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?
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.

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.

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.

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. Ett typiskt bakgrundsavsnitt är 2-3 sidor.

3 Syfte, mål, och motivation / Purpose, aims, and motivation

Här beskriver ni i princip er problemformulering. I detta avsnitt ska framgå syfte, mål, och motivation med projektet. Dessa behöver dock *inte* vara separata underrubriker.

Syfte. Vart strävar projektet? vad är det övergripande målet, nyttan, effekterna av projektet? (t.ex. bättre koll på kosthållning, enklare planering av studier...)

Mål. Vad ska konkret levereras/utföras av projektet, för att ta oss närmare syftet?

Motivation. Varför är projektet viktigt? Vilka är det viktigt för, vilka externa intressenter finns? Hur stort är problemet, vad är följden av att det inte är löst, hur bra vore det att lösa? Vilken “lucka” i området täcker ni? Varför är er lösning bättre/annorlunda än andras?

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 och hållbarhetsaspekter, men det finns förstås flera naturliga ställen att ta upp dem (*till exempel* sektionerna 5, 7, 12 och 13, 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 etiska och hållbarhetsaspekter. Tänk på stödfrågorna för att tänka på etiska aspekter:

- 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?
- Hur ser ett samhälle ut där er lösning används i stor skala?

För hållbarhetsfrågor, prova gärna “Futures Thinking” och “Systems Thinking” (se länkar i kursmaterialet).

3.1 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.)

4 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.)

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.)

Ibland är det bra att gruppera relaterade arbeten (t.ex. appar som löser liknande problem, eller andra angreppssätt än tekniska). Ibland 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”).

UPPSALA
UNIVERSITETUPPSALA
UNIVERSITET

Figur 1 Till vänster syns universitetets logotyp i färg, till höger i svartvitt.

Försök övertyga läsaren om att ni gjort ett vettigt urval av relaterat arbete (och inte bara beskriver de första google-träffarna). Beskriv gärna hur ni gjort urvalet, och motivera det.

5 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).

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.

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

6 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?)

6.1 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.

7 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”?)

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.)

Skriv tydliga krav *som går att utvärdera*. (Hur snabbt? Hur många användare? Hur strömsnålt? eller vad som är relevant).

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

8 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 rubriker (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.

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

9 DEL x+1: Algoritm för igenkänning av stora fötter

Se avsnitt 8.

10 DEL x+2: Optimering av minnesanvändning

Se avsnitt 8.

...

11 Utvärderingsresultat / Evaluation results

Beskriv resultaten av utvärderingen, när ni tillämpar de utvärderingsmetoder ni beskrivit i avsnitt 7, 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).

12 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 3.

13 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.)

14 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.

Referenser

- [BL98] T. Berners-Lee. (1998) Cool URIs don't change. W3C. Hämtad 2018-04-26. [Online]. Tillgänglig: <https://www.w3.org/Provider/Style/URI>
- [Daw09] C. W. Dawson, *Projects in Computing and Information Systems*, 2nd ed. Harlow, UK: Pearson, 2009.
- [Daw15] C. W. Dawson, *Projects in Computing and Information Systems*, 3rd ed. Harlow, UK: Pearson, 2015.
- [Wik19] Wikipedia. (2019, Mar.) Lista över vanliga språkfel. Hämtad 2019-03-22. [Online]. Tillgänglig: https://sv.wikipedia.org/wiki/Wikipedia:Lista_%C3%B6ver_vanliga_spr%C3%A5kfel

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.

B Några tips för LaTeX-användning

Ett enkelt sätt att använda/installera LaTeX för MacOS är TexShop².

För **samarbete** 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). I det fallet är det smart att dela upp dokumentet i flera (t.ex. ett per kapitel) som sen inkluderas med `\input{kapitel}`. 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

För att formatera **URLer** bättre (så att t.ex. radbrytning blir snyggare), skriv t.ex. `\url{http://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, t.ex.

²<http://pages.uoregon.edu/koch/texshop>


```
\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}...

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.

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.

B.1 BibTeX-tips

För att hantera bibliografi (**referenser**) på ett smidigt sätt, använd BibTeX! (se http://en.wikibooks.org/wiki/LaTeX/Bibliography_Management#BibTeX och nedan om referenser.)

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.

Ett verktyg för att hantera BibTeX-filer i MacOS är BibDesk³.

C Referenser

Se också kap 8.5 i Dawson [Daw15].

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.

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 `@IEEEtranBSTCTL`, se `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 [Daw15].

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

³<http://bibdesk.sourceforge.net/>

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...”.

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 [Daw15, sid 211–214], 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).

För mer info om vilken info som behövs för olika typer av referenser, se avsnitt 8.5.3 i Dawson [Daw15, Daw09]. (För att referera till flera saker samtidigt (som nyss) skriver man flera BibTeX-nycklar i samma \cite.)

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

Använd inte direktcitat, såvida inte den exakta formuleringen är viktig. Skriv hellre ett referat av vad någon sagt. (Se Dawson [Daw15, Daw09].)

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 som en start för att läsa om något, men ansträng er att hitta stabilare referenser när ni skriver om detta något. Kolla vilka referenser Wikipedia-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 [BL98] (se `refs.bib`).

D Formler, figurer, bilder, kod

Formler, figurer 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.

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

- Det är OK att skriva “Vi”!
- **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).
- **Undvik talspråk** “så”, “två stycken saker”, “ifrån”, “utav”, “var”, “kommer göra/vara” (istället för “kommer att göra/vara”, ... **Kolla på Wikipedia-sidan** “Vanliga språkfel” [Wik19] (länk i vänsterkanten i SP).
- Undvik värderande uttryck som enkelt, uppenbart.
- 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.
- 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.
- 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.
- **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.
- 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. Det är bättre att förklara begreppen när de behöver förklaras (se ovan).
- 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.”
- **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).
- 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.
- 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.

- 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!
- De av er som skriver på engelska: ni MÅSTE använda korrekta verbformer beroende på om subjektet är en eller flera saker (“it has” men “they have”).
- Undvik att skriva så här långa punktlistor, och punktlistor med så här mycket text.