

RAPPORT RAPPORT

Distribusjon/publisering av fri programvare Noen erfaringer med et slikt prosjekt

Knut W. Hansson



Rapporter fra Høgskolen i Buskerud

Nr. 84

Distribusjon/publisering av fri programvare

Noen erfaringer med et slikt prosjekt

Av

Knut W. Hansson

Hønefoss 2011

HiBus publikasjoner kan kopieres fritt og videreformidles til andre interesserte uten avgift.

En forutsetning er at navn på utgiver og forfatter(e) angis- og angis korrekt. Det må ikke foretas endringer i verket.

ISBN 978-82-8261-008-7

ISSN 0807-4488

Innholdsfortegnelse

1. Sammendrag.....	5
2. Innledning.....	7
a. Hva er fri programvare.....	7
b. Betydningen av fri programvare	8
c. Hvem lager fri programvare.....	10
d. Høgskolens interesse.....	11
e. Min interesse	11
3. Metode/design.....	13
4. Resultater.....	15
a. Mitt program	15
b. Valg av programmeringsspråk og verktøy	16
c. Fogels råd for fri programvare prosjekter	16
d. Fogel: Se deg rundt	17
e. Fogel: Velg et godt navn	18
f. Fogel: Lag en klar "Mission statement" og gjør det klart at det er fri programvare..	19
g. Fogel: Lag god liste over funksjonalitet og krav	20
h. Fogel: Angi utviklingsstatus	20
i. Fogel: Ønsker for nettsted	21
j. Fogel: Utviklingsregler	22
k. Fogel: Brukerdokumentasjon	24
l. Fogel: Utviklerdokumentasjon.....	24
m. Fogel: Eksempler på output og skjermbilder	25
n. Fogel: Bruk nettsted for fri programvare prosjekter	26
o. Publisering	27
5. Konklusjon og diskusjon.....	31
6. Appendiks – Detaljer til enkelte kapitler.....	33
Ad Fogel: Se deg rundt.....	33
Ad Fogel: Velg et godt navn	34
Ad Fogel: Lag en klar "Mission statement"	34
Ad Fogel: Lag god liste over funksjonalitet og krav	34
Ad Fogel: Angi utviklingsstatus.....	35
Ad Fogel: Brukerdokumentasjon	36
Ad Fogel: Utviklerdokumentasjon	36
Ad Fogel: Eksempler på output og skjermbilder	38
Ad Fogel: Bruk nettsted for fri programvare prosjekter.....	38
Ad Publisering.....	40
Ad Konklusjon og diskusjon	44

1. Sammendrag

Denne rapporten tar primært for seg noen erfaringer som jeg har gjort med å lage og distribuere et fritt program (fri programvare). Høgskolen i Buskerud har en spesiell interesse i dette som deleier i Nasjonalt kompetansesenter for fri programvare. Min fagseksjon (som underviser flere Bachelorprogrammer i IT) og jeg selv personlig, har også interesse av dette. Dessuten mener jeg at dette bør ha generell interesse for dem som veileder, underviser eller selv tenker å involvere seg i frie programvareprosjekter.

Med fri programvare, menes det her edb-programmer som distribueres med en fri lisens, dvs. en lisens som oppfyller kravene i Free Software Foundation. Bl.a. innebærer det at brukeren fritt kan anvende, endre og videredistribuere programmet. Kildekoden må offentliggjøres.

Fri programvare har fått stadig mer vind i seilene, og både Staten og EU er sterkt interessert i slike programmer. Sistnevnte har undersøkt både hvem som lager slik programvare og hvorfor de gjør det. Det viser seg at svært mange enkeltpersoner er involvert bl.a. for å utvikle egen kompetanse og fordi de idealistisk ønsker å støtte idéen. Kompetansen mener de fremmer karrieren. Det er også mange store bedrifter som lager fri programvare. Frie programmer brukes også av mange store bedrifter og av det offentlige, og enkelte av dem er svært kjent, f.eks. Linux, Apache, MySQL og PHP.

Undersøkelsen er utført som en case-studie. Jeg har laget et program som jeg så har lagt ut som fri programvare. Jeg gjorde da en del erfaringer, som jeg gjengir i denne rapporten. Erfaringene er naturligvis svært subjektive: De gjelder for meg og mitt ene prosjekt. Allikevel mener jeg de har en overføringsverdi til andre, tilsvarende prosjekter.

Hovederfaringene er oppsummert på side 31. Der anbefaler jeg andre som er tenker å lage et fritt programvareprosjekt å starte med en grundig kontroll av om det allerede finnes et liknende, fritt program. Hvis et slikt finnes, er det bedre å "henge seg på" det, istedenfor å lage sitt eget. Hvis man etter denne kontrollen fortsatt vil lage sitt eget prosjekt, bør man velge et FOSPHost nettsted for publisering. Da får man mange, nyttige verktøy inkludert. Deretter kan jeg anbefale at man henter Karl Fogels råd for slike prosjekter (<http://producingoss.com>) og følger dem.

Det er relativt tidkrevende, men ikke spesielt vanskelig, å komme fra et program som man selv har og er fornøyd med, til det er ferdig publisert som et fritt programvareprosjekt.

Enkelte ting bør gjøres tidlig:

- Sjekk om noen har gjort det samme før – i så fall bør man vurdere å henge seg på det
- Velg programnavn og sikre Internett-domener
- Bestem et nettsted for publisering – det har gjerne krav som man må forholde seg til og det gjøres best før man legger for mye arbeid i forhold som ikke tilfredsstillende kravene

Noen arbeidsoppgaver gikk ganske greit og tok ikke særlig lang tid:

- Å lage logo/ikon
- Å velge lisens (man bør ikke lage sin egen)
- Å lage beskrivelse til utviklere
- Å lage bilder av skjemaene

Distribusjon og publisering av fri programvare

Andre arbeidsoppgaver var mer tidkrevende, selv om de kanskje ikke var så vanskelige:

- Å lage fornuftige kommentarer i kildekoden
- Det var enkelt å sette opp nettstedet på *SourceForge.net*, men ganske tidkrevende (det kan nok variere fra nettsted til nettsted)
- Å lage brukerhjelp, selv om den var svært enkel

Det var både vanskelig og tidkrevende å lage "Mission Statement".

Fra andre prosjekter vet jeg at det kreves mye arbeid å lage god brukerhjelp, men det har jeg ikke gjort denne gang.

Rapportens struktur

Jeg har valgt å sette inn alle referanser som fotnoter, så man skal slippe å bla for mye frem og tilbake. Videre er mange detaljer tatt ut og lagt i egen appendiks. Dermed blir selve rapporten kortere og enklere å følge.

2. Innledning

Fri programvare har fått stadig mer vind i seilene de senere år. Det er mye fri programvare å finne på Internett, og det er svært mange ”prosjekter” som utvikler slik programvare. Noen av programmene er enkle og små. Andre er komplekse systemer. Noen brukes lite og noen mye. Noen er *virkelig* store som f.eks. MySQL og Linux.

Mange har kompetanse i slik utvikling (det er mange hundre tusen utviklere over hele verden som deltar på en eller annen måte), men ved min fagseksjon på høyskolen er kompetansen på dette feltet mangelfull. Høgskolen i Buskerud er medeier i Nasjonalt kompetanseseter for fri programvare, og er således interessert i at det gjøres utvikling, undersøkelser og forskning på dette feltet.

Det kan følgelig være interessant å se hva som kreves for å lage og distribuere fri programvare, helst slik at andre som ønsker det, kan delta i et utviklingsprosjekt av denne typen.

I denne rapporten gjengir jeg de erfaringer jeg selv gjorde i et forsøk på å skape et fri programvare utviklingsprosjekt.

a. Hva er fri programvare

Free Software ble definert av Richard Stallman ca 1986. Et utdrag av den seneste versjonen av definisjonen¹ sier bl.a. at

“Free software” is a matter of liberty, not price. To understand the concept, you should think of “free” as in “free speech,” not as in “free beer.”

Mer presist, er det fire friheter, nummerert fra 0 til 3, som er essensielle (*ibid*):

Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software. More precisely, it means that the program's users have the four essential freedoms:

- *The freedom to run the program, for any purpose (freedom 0).*
- *The freedom to study how the program works, and change it to make it do what you wish (freedom 1). Access to the source code is a precondition for this.*
- *The freedom to redistribute copies so you can help your neighbor (freedom 2).*
- *The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.*

Fordelen for brukerne fremgår av disse frihetene: For det første er programmet gratis, og det vil forbli gratis i fremtiden. For det andre får brukerne tilgang til kildekoden, så de kan sjekke hva som er gjort, hvordan det er gjort og de kan fritt endre det. For det tredje kan de distribuere det fritt. Og for det fjerde kan de distribuere sin endrede versjon fritt til andre.

All programvare er skrevet av noen, og er et åndsverk. Det er følgelig knyttet opphavsrett til programmet². Når man vil legge ut et fritt program, kan man da i prinsippet gi fra seg opphavsretten og distribuere programmet under det som kalles "Public Domain". Problemet er

¹ <http://www.gnu.org/philosophy/free-sw.html>

² Lov om opphavsrett, <http://www.lovdata.no/all/nl-19610512-002.html>

Distribusjon og publisering av fri programvare

at andre da kan overta programmet, videreutvikle det og selv beholde det som proprietær programvare. Det ønsker utviklerne vanligvis ikke.

Det er derfor vanligere å beholde opphavsretten, men gi gratis lisens. Lisensen vil gi klare bestemmelser om hvilke rettigheter og forpliktelser som følger med. Dette blir omtrent som å gi en gave på visse vilkår. Man kan lage sin egen lisens (som rettighetshaver kan man fritt bestemme lisensvilkårene), men det er tungt, vanskelig og kan fort bli feil. Man må jo ta hensyn til lovgivning i mange land, internasjonale avtaler osv. Derfor vil de fleste bruke en lisens som er gjennomarbeidet og har de vilkår man ønsker.

Free Software Foundation (FSF) har arbeidet mye med slike lisenser og gjort dem fritt tilgjengelige (også lisensvilkår er jo åndsverk). Hovedlisensen kalles "GNU GPL", men det finnes flere varianter og versjoner av den³. FSF har også en liste⁴ over andre lisenser som tilfredsstillers FSFs krav til en "Free License". Den vanligste av de siste er "BSD", opprinnelig fra University of California, Berkley, for operativsystemet Unix.

Når man skal distribuere fri programvare, er det en fordel å benytte en lisens som er kjent for de fleste. "GNU GPL" er da et naturlig valg. Seneste versjon har nummer 3.

Siden det ikke er noen egentlig leverandør av fri programvare, så leveres de alltid uten garantier, og det er ikke gitt at det er mulig å få støtte ("support"). Det er heller ikke sikkert at programmet vil bli vedlikeholdt. Mange frie programmer lages av en gruppe interesserte (kalt "community"), så det er viktig at man velger programmer som har en aktiv utviklergruppe, eller at man selv ser seg i stand til å vedlikeholde det. I en tidligere rapport⁵ har jeg omtalt Business Ready Rating (BRR) som er et forslag til standard for vurdering av fri programvare.

b. Betydningen av fri programvare

Den internasjonale betegnelsen for fri programvare er idag gjerne "Free Open Source Software" eller FOSS. EU har interessert seg sterkt for fri programvare og kaller det "Free/Libre Open Source Software" altså FLOSS. (EU mener at ordet "Libre" klargjør hva som skal legges i begrepet Free.)

I denne rapporten bruker jeg selv betegnelsene "fritt program" og "fri programvare". Disse betegnelsene synes å ha fått gjennomslag i Norge.

EU-kommisjonen fikk i 2006 utredet fri programvare (FLOSS). Da fant man bl.a.⁶ at *Looking at the impact of FLOSS on European competitiveness in ICTs, the study finds that such software is of great importance to the digital industry in Europe and several other parts of the world. It has already reached considerable market share in several fields, including web servers and operating systems. A large share of public and private sector organizations use at least some FLOSS.*

Findings include:

- *FLOSS applications are top rung products in terms of market share in several markets.*

³ Omtale av GNU-lisenser: <http://www.gnu.org/licenses/licenses.html>

⁴ Andre, "godkjente" frie lisenser: <http://www.gnu.org/licenses/license-list.html>

⁵ Knut W: Hansson: "LMS – Blackboard vs Moodle", Rapport nr 67 fra Høgskolen i Buskerud, 2008.

http://brage.bibsys.no/hibu/bitstream/URN:NBN:no-bibsys_brage_8136/1/6708hansson.pdf

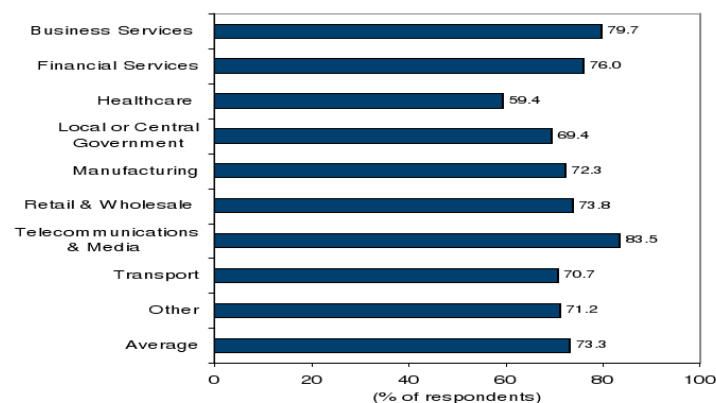
⁶ <http://www.flossimpact.eu/>

Distribusjon og publisering av fri programvare

- *The existing base of quality FLOSS applications with reasonable quality control and distribution would cost firms almost Euro 12 billion to reproduce internally. This code base has been doubling every 18-24 months over the past eight years.*
- *The notional value of Europe's investment in FLOSS software today is Euro 22 billion (36 billion in the US) representing 20.5% of total software investment (20% in the US)*
- *While the US has an edge in large FLOSS-related businesses, Europe is the leading region in terms of globally active FLOSS software developers, and leads in terms of global project leaders, followed closely by North America. Asia and Latin America face disadvantages at least partly due to language barriers, but may have an increasing share of developers active in local communities.*
- *By providing a skills development environment valued by employers and retaining a greater share of value addition locally, FLOSS can encourage the creation of SMEs and jobs.*
- *Defined broadly, FLOSS-related services could reach a 32% share of all IT services by 2010, and the FLOSS-related share of the economy could reach 4% of European GDP by 2010.*
- *Though FLOSS provides ample opportunities for Europe, it is threatened by increasing moves in some policy circles to support regulation that seeks to protect old business models of creative industries, making it harder to develop new ways of doing business.*

Vi ser at denne rapporten fremholder den store betydningen av fri programvare (22 milliarder Euro investeres årlig hvilket er 1/5-del av programvareinvesteringer, kanskje 1/3-del av alle IT-tjenester i 2010), kvaliteten på slike programmer og den raske veksten (dobling hver 18. til 24. måned). Nedenstående figur fra hovedrapporten⁷ viser hvor mange prosent som i 2005 svarer at de bruker frie programmer (FLOSS) i sitt firma:

Figure 3: FLOSS usage in Europe by industry.



Source: IDC's 2005 Western European Software End-User Survey (N=625)

Totalt svarer altså nesten $\frac{3}{4}$ at de bruker fri programvare i sin bedrift. Det er grunn til å tro at andelene er enda høyere i dag.

⁷ https://docs.google.com/viewer?url=http://ec.europa.eu/enterprise/sectors/ict/files/2006-11-20-flossimpact_en.pdf

c. Hvem lager fri programvare

Ifølge EU-rapporten (*ibid*), deltar de fleste av deltakerne i fri programvare prosjekter (78 %) på individuell, frivillig basis for å utvikle egen kompetanse. Arbeidsgivere vet å sette pris på denne kompetansen, som arbeidstakerne utvikler uten at arbeidsgiverne må betale for kurs. En stor andel (67 %) fortsetter da også å bidra til frie programmer også etter at de har lært mye, for å ”dele den kompetanse de har fått på denne måten” (en form for lojalitet) og mange (30 %) mener at de bedrer sine jobbmuligheter. Kompetansen gjelder ikke bare programmering, men også åndsverklover, evne til å samarbeide (elektronisk) i tildels meget store grupper, evne til å formulere ønsker om funksjonalitet og meget annet. Mange arbeidsgivere gir da også tillatelse til at medarbeiderne kan jobbe med frie programmer i arbeidstiden.

Noen av bedriftene som aktivt deltar i utvikling av fri programvare er faktisk hovedaktører i IT-bransjen, som IBM, Oracle, Sun og Google. Viktige unntak er naturligvis Microsoft og Apple, og de kritiseres da også av mange for det.

EU-rapporten (*ibid*) fremholder bl.a.:

FLOSS has rapidly shifted from a model driven purely by the developer community and university support to one where a main driver is industry. Not only has open source spawned a (usually) thriving environment of small focused businesses, typically devoted to development and support of specific products or to maintenance and integration activities, it has also reshaped the business models and affected the strategies of large firms, including such major industrial players as IBM, Oracle, Philips, Nokia and SAP.

Det er kanskje typisk at store firmaer inkluderer dem som allerede på forhånd definerte seg som service- og konsultentselskaper mere enn programvareprodusenter (IBM, Google). Videre kan det være naturlig at selskaper som leverer programmer som krever betydelig lokal tilpasning er med (Oracle, SAP). Endelig er det ikke unaturlig at vi her finner selskaper som primært leverer maskinvare, altså der programvaren bare er en del av produktet og ubrukelig uten maskinvaren (Phillips, Nokia).

Mer overraskende er det kanskje at også mindre programvarefirmaer står bak fri programvare. De får inntekter som konsulenter med programvaren som spesialfelt, gjennom kursing, støtte, installasjon, tilpasning og integrasjon med andre programmer. Som en foredragsholder på en konferanse sa det:

Før måtte jeg først lage et ordentlig produkt, så kunne jeg selge det. Nå legger jeg ut produktet før det er ferdig og får hjelp (eller betalt) av kundene mine til å forbedre, rette og vedlikeholde det – selv er jeg konsulent på produktet.

En annen måte som frie programmer kan gi inntekter på, er ved såkalt dual lisens. De som ønsker det, kan betale for en kommersiell lisens, som gir dem rett til å inkludere programvaren i kommersielle produkter. Den kommersielle lisensen gir også rett til støtte, nye versjoner osv. Det mest kjente programmet med en slik lisenspolitikk er antakelig databasesystemet MySQL som nå eies av Sun. Det er et av verdens mest brukte databasesystemer med mer enn 100 millioner nedlastinger totalt.

d. Høgskolens interesse

Høgskolen i Buskerud (HiBu) samarbeider med Nasjonalt kompetansesenter for fri programvare, eller Friprogsenteret. Det beskriver seg selv slik⁸:

Friprogsenteret er et uavhengig, statlig finansiert kompetansesenter for fri programvare, og bistår med å ta i bruk fri programvare, eller utvikle fri programvare i IT-løsninger. Vår målsetning er å etablere en kultur for deling, gjenbruk og samarbeid av IT-løsninger og digitalt innhold.

Friprogsenteret tilbyr arenaer for deling, bygger faglige nettverk og sprer informasjon gjennom foredrag, netjtjenester og Friprogmagasinet. Best av alt, våre tjenester er gratis.

Friprogsenteret finansieres av Fornyings- og administrasjonsdepartementet, samt Kunnskapsdepartementet.

Friprogsenteret har kontorer på Papirbredden i Drammen, sammen med høyskolen som også er én av eierne, og høyskolens dekan Hans Anton Stubberud er medlem av styret.

Man skulle kanskje tro at HiBu dermed var sterkt interessert i selv å bruke fri programvare, men i praksis er det ikke slik. Jeg skal ikke her komme inn på hva det kan skyldes. Selv bruker jeg imidlertid mye fri programvare, og har gode erfaringer med det.

I undervisningen på studiet "IT og Informasjonssystemer" bruker vi mange frie programmer. Våre erfaringer med dette er generelt positive. Fordelene for studentene er at de alltid er gratis. Studentene måtte ellers kjøpe lisenser, eller laste ned "prøveversjoner" som utgår etter en tid. Studiet ville da bli dyrere enn mange andre studier. Vi mener også at det er bra for studentene å få (gode) erfaringer med frie programmer. Disse erfaringene kan de ta med seg ut i de virksomhetene de senere skal arbeide i.

Det er også noen studenter som ønsker å lage fri programvare som sin avsluttende Bachelor-oppgave. De har behov for veiledning om hvordan det kan gjøres, og hva det krever – både hvilke arbeidsoppgaver det involverer og hvilken innsats det krever. Denne rapporten kan kanskje bistå dem med kunnskap om det.

e. Min interesse

For meg er det kommersielle av liten personlig interesse. Jeg lager programmer både for meg selv på fritiden, for å skaffe meg kunnskap om nye/endrede programmeringsspråk (egenutvikling) og som ledd i undervisningen. Mine programmer distribuerer jeg fritt – også kildekode – til andre, særlig til studentene, uten at vi noen gang diskuterer lisens. Det er typisk at mye utvikling av frie programmer tidligere foregikk på universiteter og høyskoler, eller i små, spesielt interesserte grupper. I dag er fri programvare en større industri. Siden de som distribuerer frie programmer ikke kan ta betalt, kan man undre seg over at så mange gjør det.

I vår fagseksjon ved høyskolen har vi naturligvis mye IT-kompetanse og har ingen vanskeligheter med å forstå hvordan programvare lages og vi kan gjøre det selv. Det er ingen forskjell i så måte på fri og annen programvare. Forskjellen ligger i distribusjonen og lisensieringen. Ingen av oss i dette miljøet har distribuert fri programvare. Det stiller spesielle krav, og bør være interessant både for oss ved IT-seksjonen og for høyskolen, bl.a. fordi:

- høyskolen viser interesse for fri programvare gjennom sin deltakelse i Friprogsenteret

⁸ <http://www.friprog.no/friprogsenteret/>

Distribusjon og publisering av fri programvare

- fagseksjonen benytter fri programvare både selv og i undervisningen
- som fagpersoner i IT bør vi ha kjennskap til en distribusjonsform som blir stadig vanligere
- som statlig ansatt er det interessant for meg at Staten ønsker å spre bruken av fri programvare – dette har vært et politisk satsningsområde i flere år og er det fortsatt⁹

I denne rapporten beskriver jeg hva jeg har gjort, hvordan jeg gjorde det og hvilke erfaringer jeg gjorde i dette arbeidet.

⁹ Se f.eks. <http://www.regjeringen.no/nb/dep/fad/aktuelt/nyheter/2009/fortsatt-satsing-pa-fri-programvare.html?id=581586>

3. Metode/design

For å få erfaring med å distribuere fri programvare, har jeg først laget et program som jeg bruker selv daglig, og som jeg mente kanskje kunne være interessant for andre enn meg selv.

Deretter har jeg undersøkt hvilke råd som gis hos forskjellige kilder om hva man bør passe på når man lager et fritt programvare prosjekt. En svært god oversikt er gitt av Karl Fogel i en bok som også (i tråd med "the true spirit") er lagt ut i sin helhet på Internett¹⁰. Jeg gjennomførte så alle de rådene jeg fant, og gjengir her mine erfaringer. De er gjengitt så ærlig jeg kan.

Mens Fogel er *normativ* ("*slik bør du gjøre...*") har jeg vært *deskriptiv* og konkret beskrevet mine subjektive opplevelser da jeg fulgte rådene. Siden mine erfaringer er basert på ett, eneste prosjekt der jeg selv har vært aktiv part, kan jeg ikke generalisere. Isteden har jeg derfor beskrevet hva jeg gjorde og hvordan jeg – subjektivt – opplevde det ("*slik gjorde jeg...*", "*slik ble resultatet...*" og "*dette erfarte jeg...*").

Slik sett har denne undersøkelsen preg av å være *intensiv, kvalitativ, deskriptiv og eksplorativ*. På en måte kan man si at denne undersøkelsen er en case-studie, men siden jeg selv er svært aktiv deltaker, vil jeg heller beskrive det som *aksjonsforskning* (der forskeren selv deltar). På denne måten har jeg ikke noe grunnlag for å generalisere, men får på den annen side selv svært intim innsikt i problemstillingene.

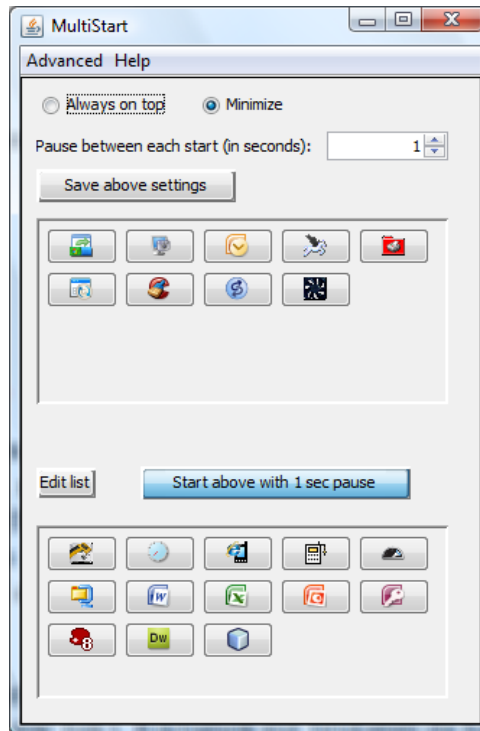
Man bør her være oppmerksom på at mine erfaringer er personlige, og gjelder ett, spesielt program. Jeg kunne hatt andre erfaringer med et annet – f.eks. et større – program. Andre personer ville nok også ha fått andre erfaringer med mitt program. Erfaringsdelen av denne rapporten er altså nødvendigvis *subjektiv*. Jeg håper allikevel at mine erfaringer kan være nyttige for andre.

¹⁰ Karl Fogel: "Producing Open Source Software - How to Run a Successful Free Software Project", Fogel, 2009. Boken kan kjøpes i papirutgave, men kan også finnes i sin helhet på Internett (<http://producingoss.com/>). Boken er publisert under Creative Commons lisens (<http://creativecommons.org/licenses/by-sa/3.0/>).

4. Resultater

a. Mitt program

Programmet mitt er et hjelpeprogram for å starte andre programmer. Hovedskjemaet ser slik ut (her i halv størrelse):



Øverst er det et par menyer og noen valg for standard visning. Videre er det to rammer med knapper for et antall programmer. I disse rammene har jeg samlet knapper for programmer som jeg bruker mye, men ikke ønsker startet ved innlogging. Jeg kan starte ett og ett program ved å klikke på den tilhørende knappen. Dessuten kan jeg starte alle programmene i den øverste rammen med ett klikk på knappet "Start above...". Det gjør jeg ofte – men ikke alltid – ved oppstart av maskinen. Jeg har sørget for at programmet mitt startes automatisk med pålogging, så jeg kan velge hva jeg vil gjøre. Det er også enkelt å hente dette programmet frem, hvis jeg vil starte ett av programmene.

Brukeren bestemmer hvilke knapper som skal stå i hver av rammene.

Det viktigste for meg, er at jeg med ett museklikk kan starte mange programmer, og at jeg selv kan bestemme hvilke programmer det skal være. Jeg er også opptatt av at programmene kan startes hver for seg, om jeg skulle foretrekke det.

Programmet kan lastes ned¹¹ fra prosjektets hjemmeside¹² på Sourceforge.net, og krever ingen installering.

¹¹ Direkte nedlasting: <http://sourceforge.net/projects/multistart/>

¹² Prosjektets hjemmeside: <http://multistart.sourceforge.net/>

b. Valg av programmeringsspråk og verktøy

Når man skal lage et fritt program, må andre programmer som inkluderes også være fri. Hvis man f.eks. bruker en Microsoft Server database, må jo brukeren kjøpe lisens for den. Hvis man inkluderer deler som er proprietære, må man betale lisens til rettighetshaveren. Det vil ødelegge modellen for fri programvare.

Verktøy som brukes, bør også være fri av hensynt til andre som vil delta. De bør også fritt kunne anvende de samme verktøyene.

Mitt prosjekt

Jeg har brukt *Java* som programmeringsspråk og *NetBeans* som programmeringsmiljø. Begge deler er fri programvare (som Sun står bak). Riktignok må brukeren da ha Java Run Time System installert på sin maskin, men det har svært mange allerede og det er uansett fritt å redistribuere det sammen med sin egen programvare. Jeg måtte dessuten også bruke *Filezilla* til opplasting til nettstedet, og det er fri programvare.

Programmets logo er laget med programmet *IconEdit*. Til bilde av programmets skjemaer, brukte jeg *Gadwin PrintScreen*. Videre har jeg benyttet billedbehandlingsprogrammet *Paint Shop Pro*. Disse er ikke fri programvare (men jeg har min egen lisens). Da de ikke inkluderes i min applikasjon, så er lisensen uansett ikke noe problem.

Mine erfaringer

Før dette arbeidet begynte, hadde jeg programmet ferdig, programmert med Visual Basic. Når programmet skulle distribueres fritt, og for å gjøre programmet kjørbart for flere operativsystemer, måtte det skrives om i Java. Java er ikke mitt sterkeste programmeringsspråk, selv om jeg har programmert mye objektorientert. Jeg brukte følgelig mer tid på omprogrammeringen enn jeg nok hadde tenkt, og det ble flere ting jeg ikke fikk til. Det kan være at det ikke er mulig, eller så har jeg kanskje for liten kompetanse i språket. Særlig skapte det grafiske brukergrensesnittet problemer. Visual Basic er tett integrert og tilpasset Windows operativsystem og det er enkelt å få til "snertne", grafiske funksjoner. Java har naturlig nok ikke den samme tilpasningen til noe bestemt operativsystem.

Jeg oppdaget også at jeg ble sterkt påvirket av at programmet skulle distribueres og vises for "all verden". Av hensyn til min egen "ære", ble det forbausende viktig for meg at ting var gjort riktig, virket som de skulle, hadde ryddig kode osv. Det siste er jo uansett viktig hvis andre skal kunne endre koden.

Valget av programmeringsmiljøet NetBeans gjorde det relativt enkelt å lage det grafiske grensesnittet, men NetBeans lager mye kode som er vanskelig å følge og ekstra filer som anvendes under kompilering. Derfor er det komplisert å endre programmet med andre verktøy. De som vil endre mitt program, er nok i praksis også tvunget til å bruke det samme programmeringsmiljøet. Det synes jeg er uheldig. Jeg ser i ettertid at jeg heller burde skrevet all kode for grensesnittet selv – "den tunge veien".

c. Fogels råd for fri programvare prosjekter

I boken "Producing Open Source Software" nevnt ovenfor, gir Karl Fogel i kapittel 2 systematiske råd om hvordan man kommer i gang med et friprogram prosjekt. Her skal jeg gi rådene videre, og si hva jeg selv gjorde med dem. Karl Fogel har utviklet fri programvare i en rekke store prosjekter i 15 - 20 år. Han har også skrevet flere, kjente bøker om forskjellige sider av dette emnet.

Distribusjon og publisering av fri programvare

Fogel starter med å understreke at fri programvare må oppnå interesse fra to grupper mennesker for å få suksess:

1. Brukere
2. Utviklere

Disse to gruppene er svært forskjellige – de har delvis behov for samme informasjon, men mye informasjon er også spesiell for hver av dem. Det er viktig, iflg. Fogel (ibid.), at alt som publiseres gjør det helt klart hvem informasjonene er myntet på. Derved unngår man at brukere irriteres av detaljer som ikke interesserer dem, og at utviklere ikke finner informasjon som er viktig for dem. En lenke til ”dokumentasjon” er da en dårlig benevnelse, fordi den ikke gjør det klart om det er bruker- eller utviklerinformasjon det er snakk om.,

Videre er det alltid viktig med utseende, navn, logo og andre ting som bidrar til markedsføring.

Nedenfor ser jeg på Fogels råd og forteller om mine erfaringer.

d. Fogel: Se deg rundt

Fogel mener at det er viktig å kontrollere at det ikke allerede finnes et annet prosjekt som gjør omtrent det man selv ønsker. Kanskje finner man ikke akkurat det samme, men noe som likner så meget at man heller kan involvere deg i det andre, med sikte på å utvide funksjonaliteten av det. Man kan f.eks. søke på Freshmeat (en nyhetsformidler for fri programvare), sourceforge (det største nettstedet for fri programvare prosjekter) og Free Software Foundation¹³. For egen del vil jeg legge til Google (eller tilsvarende). Den finner mye som ikke ligger på de nevnte nettsidene.

Mitt prosjekt

Jeg gjennomførte de nevnte søkene, inkludert søket med Google. Jeg fant mange, liknende programmer som mitt. Det er flere typer:

1. ”Dashboard” som viser et lite vindu fast på skjermen. Ikoner gjør det mulig å starte et gitt program raskt. Jeg fant ingen slike som kunne starte flere programmer med et klikk.
2. Programmet *LaunchIt Now* dukker opp ved behov og har knapper for oppstart (som mitt), men mangler muligheten til å starte flere programmer med ett tastetrykk. Jeg har brukt det i en tidligere versjon, og var da svært fornøyd. Det er gratis, men ikke fri programvare.
3. Tekstbaserte kommandovinduer som indekserer Start-menyen. Mens man skriver, leter programmet etter en match på startmenyen (og evt. andre kataloger), og viser ikon og fullt navn på den. Enter-tasten starter da programmet. Dette programmet har et annet formål enn mitt.
4. Programmer som knytter bestemte tastekombinasjoner (f.eks. CTRL+ALT+L) til start av et bestemt program. Problemet da er jo å huske kombinasjonen og man må starte programmene ett og ett.
5. Program som gjør det mulig å forsinke starten av et oppstartsprogram med noen sekunder. Dermed unngår man at oppstartsprogrammene konkurrerer om maskinressursene, og oppstarten går raskere (og med mindre problemer – noen ganger ødelegger ett oppstartsprogram for et annet på min maskin).

¹³ Henholdsvis <http://freshmeat.net>, <http://www.sourceforge.net> og <http://directory.fsf.org>

Distribusjon og publisering av fri programvare

Ett program – *Run It All* – likner svært på mitt¹⁴. Det starter – som mitt – flere programmet samtidig med ett museklikk. Programmene som startes, finnes på en tekstbasert liste (jeg bruker grafiske knapper, hvilket gjør det enklere å starte bare ett av programmene). *Run It All* er gratis men ikke fri programvare. Jeg har forsøkt å installere programmet for å prøve det, men fikk det ikke til.

Det finnes også et gratis program (åpen kildekode, men ikke fri programvare) som likner *svært* på mitt. Det heter *MultiRun*¹⁵ (likner også på det navnet jeg har valgt). Programmet kjører kun under Windows. Jeg har forsøkt å installere og prøvekjøre, men får bare feil (programfeil, påstås det). Det lover jo ikke helt godt for programkvaliteten.

Mine erfaringer

Jeg erfarte altså at mange har sett samme problem som meg, og har gjort noe med det, på den ene eller den andre måten. Jeg fant imidlertid ingen som gjør akkurat som mitt program, men minst to likner svært. Det kan allikevel tenkes at det er bruk for ”enda ett slikt program” (mitt!).

Videre erfarte jeg at dette bør gjøres tidlig. En grundig sjekk kan vise at det allerede finnes et program som er som det man selv tenker å lage. Da kan man spare seg arbeidet og heller delta i det andre, eksisterende prosjektet.

e. Fogel: Velg et godt navn

Navnet på prosjektet er viktig – det er det første andre ser. Det står gjerne i overskrifter overalt hvor programmet omtales. Det bør altså si noe vesentlig om hva programmet faktisk gjør. Dessuten bør det være lett å huske, eller lett å komme på når man vet hva programmet gjør.

Engelsk er de facto blitt det foretrukne språk for Internett og programvare. Navnet bør altså være engelsk, hvis man ikke sikter bare rent lokalt på et norsk publikum. Man bør unngå vitser og annet som krever intimt kjennskap til engelsk – mange av brukerne har ganske dårlige engelskkunnskaper.

Fogel anbefaler at man sjekker varemerker. Ett slikt register er det amerikanske Trademark Electronic Search System, TESS¹⁶. Et internasjonalt søkested er World Intellectual Property Organization, WIPO¹⁷.

Videre anbefaler Fogel at man sjekker tilgjengeligheten av domenet med samme navn – i første omgang *org*. Det kan være smart også å registrere på *com* og på *net* slik at disse ikke forblir ledige til andre. De to siste bør bare henviser til *org* domenet.

Jeg kan ikke se at Fogel nevner *logo* og det finner jeg litt overraskende.

Mitt prosjekt

Opprinnelig hadde jeg kalt programmet ”Oppstart”. Da programmet ble gjort engelskspråklig, skiftet jeg navn til ”*StartUp*”. Det viste seg imidlertid at dette navnet allerede var i bruk (for et program som redigerer hvilke programmer som skal startes ved maskinstart). Både i TESS

¹⁴ Se skjermbilde side 28.

¹⁵ Se skjermbilde side 28.

¹⁶ <http://tess2.uspto.gov/>

¹⁷ <http://www.wipo.int/ipdl/en/madrid/>

Distribusjon og publisering av fri programvare

og i WIPO fant jeg *StartUp*, men ikke *MultiStart*. Jeg forsøkte da mange andre navn, uten særlig hell. (Som et kuriosum kan jeg nevne at "*StartIT*" viste seg å være opptatt av et virus! I tillegg er det navnet på et IT-firma, en facebookgruppe for dette firmaet, en drill, et program for å lage presentasjoner på CD, et teknologimagasin og et IT kurs og sikker mye annet.)

Jeg endte med navnet "*MultiStart*" fordi jeg synes det dekker funksjonaliteten godt. Jeg kunne ikke finne noe annet program med samme navn, men det er en rekke andre *produkter* og *firmaer* med dette navnet.

Når det gjelder Internett-domene, så viste det seg at *multistart.com* er opptatt, mens de to andre var ledige. *Multistart.org* er eid av et firma som vil selge domenet for US\$ 2.910 – nettsiden er altså ikke aktiv. Registrering av domenet *multistart.org* vil koste ca Kr. 198 for to år. Det har jeg ikke gjort.

Jeg laget både ikon (.ico) og en logo (.png) for programmet¹⁸.

Mine erfaringer

Jeg fikk problemer med å finne et passende navn. Med så mange programmer som allerede er publisert, vil det lett være mange navn som allerede er brukt av andre. Det ville antakelig være lettere å finne et norsk navn (tilfanget av programmer med norsk navn er mindre), men man ønsker jo at programmet skal kunne brukes globalt og da bør det ha engelsk – eller et helt språkøytralt – navn. Dette arbeidet bør gjøres tidlig. Det var mye ekstraarbeid med å endre navnet etter at programmet var ferdig, fordi det brukes i skjermbilder, filnavn osv.

Å lage en logo opplevde jeg som mye enklere og ikke særlig tidkrevende.

f. Fogel: Lag en klar "Mission statement" og gjør det klart at det er fri programvare

Mission statement er en beskrivelse av programmet og/eller hva prosjektet ønsker å oppnå. Beskrivelsen må være tydelig. Det er mange eksempler på uklare beskrivelser og min erfaring er at interessen daler svært fort når man ikke helt vet hva programmet gjør. Det er nok få som vil laste ned, installere og prøve programmet uten helt å vite hva de får. Utprøving er heller for å sjekke kvalitet enn funksjonalitet.

For det første bør man gjøre det klart – allerede på nettstedets åpningside – at dette er gratis, at det er fri programvare og hvilken lisens som gjelder. Brukerne har antakelig dårlig erfaringer med "free programs" som viser seg å slutte å virke etter en tid med beskjed om at du nå må betale for kunne bruke programmet mer. Da har brukeren allerede lagt mye arbeid i oppsett, gjøre seg kjent med programmet osv, og vil føle seg lurt.

Mitt prosjekt

Mens jeg så etter programmer som kanskje gjør det samme som mitt, erfarte jeg at mange av disse programmene har nettsider som forklarer svært dårlig hva de egentlig gjør. Den tekstlige beskrivelsen er sjelden tilstrekkelig, men det er allikevel viktig.

Det var ikke helt enkelt å lage en god og kort beskrivelse av applikasjonen min. Det gjelder jo å gjøre den rimelig komplett uten at den blir for lang. Etter en god del arbeid, fikk jeg frem en "Mission Statement" som jeg var noenlunde fornøyd med¹⁹. Det er mye mer som kunne sies

¹⁸ Bilde av logoen side 29

¹⁹ Se "Mission Statement" på side 29

Distribusjon og publisering av fri programvare

om programmet *MultiStart* enn det jeg fikk med, men det gjøres nok bedre andre steder enn i denne beskrivelsen.

Som omtalt ovenfor, finnes det forskjellige lisenser for fri programvare. UNESCO anbefaler at hvis man vil velge konservativt, bør man velge programvare som dekkes av GNU GPL eller GNU LGPL lisens. UNESCO gir anbefaling til brukerne, der de bl.a. sier (*UNESCO, lisenser*):

- *Only deal with software covered by the GNU GPL or GNU LGPL and documentation covered by the GNU FDL*
- *Don't attempt to write a Free Software license*
- *Don't attempt to find out if a license is Free Software*

Dette er riktignok ikke ment som et definitivt råd, bare som en første approksimasjon. Allikevel velger jeg GNU-lisens, på grunn av den tryggheten det gir for brukerne.

Mine erfaringer

Det var relativt enkelt å velge en passende lisens (men man bør ikke forsøke å skrive en selv), men "Mission Statement" var ganske vanskelig å lage og krevde mye omtanke.

g. Fogel: Lag god liste over funksjonalitet og krav

Tanken bak dette er at leseren (en potensiell bruker) raskt kan se om dette er noe for ham. Er det dette han ser etter? Kan det kjøres på hans maskin med hans operativsystem? Dette blir da en utvidelse av *mission statement*.

Mitt prosjekt

Funksjonalitet skal si noe om hva programmet gjør, altså hva det kan hjelpe brukeren med. Krav gjelder de krav som programmet stiller til omgivelser. Særlig aktuelt er operativsystem, men det kan også være krav til at andre programmer er installert. Mitt program skal kunne kjøres på de aller fleste operativsystemer, men det krever at Java Run Time System er installert. Jeg kom etter hvert frem til en liste²⁰ med "features" og "requirements" på ni punkter. I tillegg la jeg her igjen inn henvisning til lisensen GNU GPLv3.

Mine erfaringer

Når man har laget et program, så kjenner man det jo godt. Man skulle da tro at det var enkelt å lage en beskrivelse av funksjonalitet og krav. Det viste seg imidlertid at det faktisk krevde en del arbeid å lage denne. Det gjelder jo å være kort, men allikevel forklarende, og det er lett å glemme enkelte ting som programmet gjør.

h. Fogel: Angi utviklingsstatus

Spørsmålet er hvor langt programmet er kommet. Programmer utvikles jo over tid, og et poeng med å distribuere fri programvare og derved skape et åpent prosjekt, er nettopp at man vil gi fra seg programmet før det er ferdig og få hjelp med videre utvikling.

Foster advarer mot "hype". Man bør ikke skryte av ting som ikke virker, kvalitet man ikke er sikker på osv. Det vil straffe seg senere, mener han, fordi programmet får et dårlig rykte. Slikt sprer seg fort på Internett og er svært vanskelig å bli kvitt senere. Man bør gjøre klart hvor langt det er mellom faktisk program som slippes og *mission statement*. Hvis man har en alfa

²⁰ Min liste er gjengitt på side 29

Distribusjon og publisering av fri programvare

eller beta versjon, bør man si det klart. Man ønsker seg vel da helst medutviklere og ikke vanlige brukere.

Det kan være aktuelt med en liste over kjente feil og ønsker for videre utvikling. For programmer med flere versjoner, er det også aktuelt med en liste som viser utviklingen over tid – hvilke versjoner har vært offentliggjort og hva var forskjellen på dem og forrige versjon? Poenget er å gi leserne en følelse av at det er ”liv” i prosjektet og at de kan forvente at kjente feil rettes og utvidelser/forbedringer følger i senere versjoner. Et program kan være greit nå, men hva skjer f.eks. i forhold til 64-bits operativsystemer, ny maskinvare osv. Vil det bli vedlikeholdt? Jeg har f.eks. selv opplevd at Lexmark ikke ville lage ny versjon av driveren for min skriver fordi den ikke lenger er i salg.

Mitt prosjekt

Jeg ser at noen slike lister skiller mellom *bugs* (feil) og *issues* (problemer). Videre skiller det mellom *missing features* (kjent, manglende funksjonalitet) og *planned features* som også kalles *enhancements* (funksjonalitet som er ønskelig men ikke direkte manglende – de vil bli føyet til i senere versjoner). Jeg kan henvise til Firefox, som har en meget bra side²¹ med slik ting.

For meg er det ikke aktuelt å gjengi programmets historie – det er første versjon som nå legges ut. Jeg mener at mitt program er rimelig ferdig til bruk (versjon 1.0). Jeg er ikke kjent med feil, men jeg ser noen muligheter for videreutvikling.

Mine erfaringer

Jeg har laget et program som jeg egentlig er ganske fornøyd med og som er i daglig bruk. Allikevel kom jeg, ved nærmere ettertanke og kritisk vurdering, at det faktisk var ganske meget som gjenstod²². Det var litt ”nedtur”. Det var allikevel positivt at det tross alt er mulig å selv se mangler ved sitt eget produkt, hvis man bare er åpen for det.

i. Fogel: Ønsker for nettstedet

Fogel fremholder mange ønsker for nettstedet der prosjektet legges ut: Mulighet for nedlastinger, versjonskontroll, feilsparing, kommunikasjon og råd, dokumentasjonens tilgjengelighet og andre ting til nettstedet.

Nedlastinger bør skje i standard formater. I begynnelsen er det kanskje ikke behov for eksekverbare koder, unntatt når det vil være vanskelig å få programmet til å kjøre uten. Poenget er jo da å få utviklere på banen, og de vil se koden og kompilere selv.

Distribusjonen må være enkel og standardisert. Jo mer ekstra som kreves, desto færre vil ta seg bryet.

Pass på at hver versjon som offentliggjøres er tydelig merket med versjonsnummer, slik at andre kan se om de har siste versjon (*build*).

Fogel mener at det er viktig å vise prosjektdeltakere at det holdes orden på feil osv. og at versjonskontroll er viktig. Man må huske på at folk kan arbeide med programmet døgnet rundt, på forskjellige steder i verden. De vil da hente seneste versjon når de skal gjøre

²¹ <http://www.mozilla-europe.org/en/firefox/3.6/releasenotes/>

²² Se min angivelse av utviklingsstatus på side 16

Distribusjon og publisering av fri programvare

endringer og tillegg. De vil også vite hva som trengs å bli gjort, og hva som er løst. Hvis man ikke lager en slik med en gang, bør man annonsere at det kommer.

Fogel mener også at mange ser på om det kommer inn en stadig strøm av feilmeldinger, forbedringsønsker o.l., fordi det forteller dem at det er ”liv” i prosjektet.

Mange vil ønske å vite hvordan de kan komme i kontakt med menneskene bak prosjektet. Det er derfor aktuelt med e-post, praterom, diskusjonsforum, postlister o.a. For den siste bør det gjøres klart at utviklerne abonnerer på listen, så de som sender inn vet at det når dem.

I begynnelsen mener Fogel at det er liten forskjell på utviklerne og brukerne, så de kan ha samme forum. Etterhvert bør man ha separate fora for de to gruppene.

Dokumentasjon bør være tilgjengelig to steder: På Internett (fra prosjektets hjemmeside) og som en del av den nedlastede pakken (da blir den tilgjengelig lokalt). Dokumentasjonen bør være i form av én, lang side. Dermed blir den søkbar f.eks. etter et bestemt ord som brukeren vet finnes, men ikke kan huske akkurat hvor sto.

Fogel lister opp mange andre ting som kan være aktuelle senere, f.eks. en nyhetsside, prosjekthistorien, lenker til andre steder, mulighet for å søke hele nettstedet og mulighet for donasjoner.

Mitt prosjekt

Jeg brukte et nettsted der alle nedlastinger er i form av tydelig merkede lenker. Valg av nettsted (det ble *sourceforge.net*) er omtalt nedenfor.

Nettstedet fikk verktøy for versjonskontroll, melding av feil og kommunikasjon i form av forum e.l. En mailinglist laget jeg også. Hjelp har jeg ennå ikke laget.

Ellers satte jeg opp mange av de verktøy som nettstedet tilbyr, f.eks. nyhetsside, prosjekthistorie og mulighet for søking på hele nettstedet. Donasjoner har jeg ikke benyttet – dette er jo et FoU-prosjekt for meg og jeg tror ikke det er verd bryet å ordne betalingsformidlingen som ville være nødvendig. Hvis applikasjonen mot formodning skulle bli voldsomt populær, så kan jo dette komme senere.

Mine erfaringer

Jeg kommer tilbake med erfaringene med dette sammen med en beskrivelse av arbeidet med å sette opp nettstedet.

j. Fogel: Utviklingsregler

Hvis noen, iflg. Fogel, tenker å være med på utviklingen, så vil de trenge litt informasjon om hvordan ting gjøres, f.eks.

- pekere til forum for interaksjon med andre utviklere
- instruksjon om hvordan feil rapporteres og rettelser og tillegg sendes inn
- indikasjon på hvordan utvikling vanligvis gjøres – er prosjektet styrt av en ”opplyst enevelde” eller er det demokrati eller hva annet.

Fogel mener ikke noe negativt med ”opplyst enevelde” (han kaller det ”benevolent dictatorship”). Han mener mange vellykkede prosjekter styres slik, men det er viktig å si det åpent, så folk vet hva de har å forholde seg til.

Fogel henviser til det han mener er et godt eksempel²³. Derfra er nedenstående klippet:

The best way to contribute to the Apache foundation is to get involved with one of its many communities.

Use the products and give us feedback

Using the products, asking (and later answering) questions, reporting bugs and making feature requests are critical parts of the project community. User feedback can strongly drive the projects and the technologies.

- [Find out about projects](#)
- [Join mailing lists](#)
- [Download releases](#)
- [Report bugs/request features](#)

Contribute back to the community

Beyond the user activities, it'd be great to see you helping with the development of the projects. This can mean getting involved with the discussions on the development mailing lists, answering user questions, providing patches for bugfixes or features, and helping to improve the documentation.

- [Join mailing lists](#)
- [Checkout the latest code](#)
- [Create bug/feature patches](#)
- [Understand How It Works](#) and see other [notes for contributors](#)

Become a committer

We'd love to see you showing the commitment that means we can make you a committer and have you join in. Here are some comments from the history books on getting involved and the open source mentality.

- [Contributing](#) – Craig R. McClanahan
- [Understanding Open Source](#) – Cameron Riley

Refer to [ASF development infrastructure](#) and [notes for committers](#).

Vi ser at det er både fornuftig tekst og mange lenker til forskjellige sider med informasjon om de tingene Fogel nevner, f.eks. mailinglist (separate for utviklere og brukere), downloads, bugs/request forum, seneste versjoner av koden, levering av patches osv. For utviklerne er det også en Wiki, bug database, tips og mye annet.

Mitt prosjekt

I alle fall i begynnelsen, har jeg holdt full kontroll på koden. Det vil si at jeg leder prosjektet og aksepterer/forkaster alle endringer – med andre et ”opplyst enevelde”. Det har jeg sagt klart fra om i dokumentasjonen til utviklere.

Jeg tror – i motsetning til Fogel – at det helst skal være et eget forum for utviklere fra start, i alle fall hvis nettstedet enkelt støtter det. Der kan man også forklare hvordan feil rapporteres og rettelser/tillegg sendes inn.

Mine erfaringer

Jeg kommer tilbake med erfaringene med dette sammen med en beskrivelse av arbeidet med å sette opp nettstedet.

²³ <http://apache.org/foundation/getinvolved.html> (Fogels referanse er ikke lenger aktuell – bruk evt. denne)

k. Fogel: Brukerdokumentasjon

Det er minst to lesere av dokumentasjonen: Brukere og utviklere. De trenger helt forskjellig informasjon. Brukerne trenger bruksanvisning for installasjon (med krav til øvrige systemer, maskin og operativsystem), bruk av applikasjonen og tips om bruken. Utviklerdokumentasjonen kommer jeg tilbake til nedenfor.

Dette er omfattende dokumentasjon og mange av disse dokumentene må endres når applikasjonen endres (konfigurasjonsstyring). Det synes det meget mot.

Fogel anbefaler at man bruker verktøy som gjør det enkelt å endre, f.eks. HTML, XML eller ren tekst – noe som er enkelt å bruke og gjør det lett å endre.

Noen tommelfingerregler for brukerdokumentasjonen:

- Fortell leseren hvilke forkunnskaper som forventes av ham.
- Forklar klart, grundig og tidlig hvordan applikasjonen installeres og hvordan applikasjonen enkelt kan testes for å se om installasjonen er blitt korrekt. Dette mener Fogel er enda viktigere enn brukerdokumentasjonen, fordi det gjelder å få brukeren til å begynne å anvende applikasjonen raskt. Hvis de først gir opp, så er det gjerne for alltid.
- Gi en klar veiledning (tutorial) i hvordan vanlige oppgaver løses med applikasjonen. Helst skulle det være flere slike, men én grundig veiledning viser i alle fall brukeren at applikasjonen kan brukes til noe nyttig – så kan han selv prøve andre ting.
- Vis klart hvor applikasjonen ikke er komplett. Da viser du brukeren at du er kjent med problemet og gir håp om at det løses senere. De kan gjerne vises som områder der utviklerne trenger hjelp, i håp om å involvere flere.

Det siste punktet mener Fogel gjelder for alt som vises i prosjektet. Ingen vil oppfatte det som at produktet er dårlig, heller at utviklerne har en viss styring og vet hva de driver med.

Fogel nevner spesielt behovet for en stadig voksende FAQ. Den må lages etter hvert som brukerne stiller spørsmål som besvares. Det finnes egne verktøy for dette.

Mitt prosjekt

Foreløpig er det ikke laget fullstendig brukerhjelp. Jeg laget en omtale av installasjon og litt om bruk. Den er lagt i en *LesMeg.txt*²⁴ fil som følger med, og den er tilpasset nettstedet der programvaren ligger.

Mine erfaringer

Fra andre prosjekter vet jeg at det kreves mye arbeid å lage god brukerhjelp. Det har jeg da heller ikke gjort, annet enn på et rudimentært nivå. Det må imidlertid komme senere, så det er bare snakk om en utsettelse. På den annen side kan jo publisering muligens gi hjelp til å lage slik dokumentasjon. Selv den enkle hjelpen som jeg laget, tok det endel tid å lage.

l. Fogel: Utviklerdokumentasjon

Utviklerne trenger systemdokumentasjon med beskrivelse av filer og database, moduler i programmet, koden, hva applikasjonen trenger av ressurser (type database, run-time system, operativsystem, utilities) og hva som brukes til utviklingen (programmeringsmiljø, IDE). Systemdokumentasjonen lages for at utviklere kan forstå applikasjonen og endre den. Det er noe litt annet enn utviklingsreglene som ble nevnt ovenfor.

²⁴ Teksten er gjengitt på side 31

I verste fall kan systemdokumentasjonen vente. De opprinnelige utviklerne kan jo svare på spørsmål som kommer, og derved bistå dem som vil sette seg inn i koden. Etterhvert blir det klart hva utviklere lurer på og systemdokumentasjonen kan sikte på å svare på nettopp disse spørsmålene.

Mitt prosjekt

Det viktigste som finnes foreløpig av systemdokumentasjon, er programkoden selv. Jeg har forsøkt å kommentere den så godt jeg kan, så en utvikler som kan Java, bør kunne sette seg inn i den. Filene som brukes inneholder også forklaring i klartekst²⁵.

Kommentarene skrev jeg på engelsk av hensyn til et internasjonalt publikum (se ”*Fogel: Velg et godt navn*” angående valg av språk). Jeg tror kommentarene vil være tilstrekkelig for en Java-programmerer, men det er vanskelig å vite. Hvis noen andre utviklere mener at det bør kommenteres mer, kan de jo legge inn en forespørsel om *enhancement*.

Jeg laget også en kort beskrivelse til eventuelle (med)utviklere²⁶.

Mine erfaringer

Å lage ovenstående beskrivelse til utviklerne, gikk relativt raskt. Jeg kjenner jo tross alt mitt eget program godt. Derimot tar det overraskende lang tid å skrive fornuftige kommentarer i kildekoden. Det er vanskelig å tenke seg hva andre programmere trenger å vite om koden, og det skal helst ikke være for mye, for da ”drukner” koden i kommentarene.

m. Fogel: Eksempler på output og skjermbilder

Skjermbilder er svært instruktive når det gjelder å forstå hva applikasjonen faktisk gjør. Dessuten er de en klar indikasjon på at den faktisk virker. Et bilde sier jo mer enn tusen ord...

Mitt prosjekt

Mens jeg så etter programmer som kanskje gjør det samme som mitt, erfarte jeg at mange av disse programmene manglet noe så enkelt som et bilde av den kjørende applikasjonen. Den tekstlige beskrivelsen er sjelden tilstrekkelig.

Man må selvsagt vente til skjemaene definitivt er klare. Det er svært irriterende når bildene viser noe annet enn det man ser når man selv kjører programmet. Jeg har i hovedsak tre skjermbilder²⁷. I tillegg kan det være meldingsbokser o.l., men dem tok jeg ikke med – jeg synes ikke at de sier noe vesentlig om hva programmet gjør.

Mine erfaringer

Med et skikkelig bildeprogram, er det enkelt å lage bilder av skjemaene. Med nyere versjoner av Windows (Vista, 7), finnes det innebygget (CTRL+ALT+PrtScr tar bilde av det aktive vinduet). Dette var følgelig fort gjort.

Det viste seg imidlertid at ett av bildene måtte endres, da det var for stort etter nettstedets regler. Også dette var fort gjort med et bildebehandlingsprogram.

²⁵ Se eksempler på side 31

²⁶ Se teksten på side 31

²⁷ Skjermbilder av mitt program er gjengitt på side 33

n. Fogel: Bruk nettsted for fri programvare prosjekter

Dette kalles gjerne "canned hosting". Det innebærer at prosjektet etableres på en egen portal for fri programvare prosjekter.

Fordelen ved slike portaler er at de har med mange verktøy som er interessante for et fritt programvare prosjekt. De tilbyr forskjellige, så avhengig av behov, kan det være aktuelt å velge mellom dem. Mange av dem er gratis – i alle fall for en viss mengde data.

Ulempen er at man får mindre kontroll med utseende og verktøy. F.eks. kan de vise reklame (de skal jo også overleve) som kan virke forstyrrende. Det er derfor ikke helt uvanlig å inkludere en lenke til eget nettsted som er bygget opp "lekrere".

Mitt prosjekt

Wikipedia har en meget god oversikt²⁸ – laget av mange i samarbeid – over nettsteder for fri programvare, som de kaller "Open Source Software Hosting Facilities". Andre legger til "Free" og "Project" får "Free/Open Software Projects Hosting" som gir forkortelsen "FOSPHost". Jeg synes den siste betegnelsen er den mest komplette og kaller det her for *FOSPHost nettsted*.

Oversikten over FOSPHost nettsteder er meget omfattende, og det er angitt svært mange egenskaper ved dem. Jeg foretok derfor først en "grovsiling". Da stilte jeg fem absolutte krav som nettstedet måtte oppfylle ("knock out factors"), f.eks. at nettstedet skulle ha eksistert noen år og ha solide eiere²⁹. Listen over aktuelle nettsteder ble da sterkt redusert.

Videre satt jeg opp en liste over egenskaper som er ønskelige³⁰, men ikke noe krav. Jeg vil altså foretrekke nettsteder som har disse ønskede egenskapene. Nettstedene har også mange andre egenskaper³¹ som ikke betyr noe for meg, f.eks. om visse land er ekskludert (utenom Norge) og om nettstedet selv bruker fri programvare.

Jeg går ikke i detalj om alle de FOSPHost nettsteder som er vurdert, men etter en grovsiling utfra ovenstående kriterier står jeg igjen med bare to nettsteder:

1. **Berlios**. Denne får jeg ikke tilgang til. Nettstedet støttes/drives av et tysk forskningsinstitutt. Det sies å ha vært utsatt for hacking tidligere i år, og kan muligens derfor være stengt.
2. **SourceForge**.

Av spesiell interesse for høyskolen (men ikke for meg denne gang) er kanskje **OSOR.eu**, som er støttet av EU-kommisjonen og bare er for offentlige administrasjonsprosjekter.

Til syvende og sist var det bare *SourceForge.net* som tilfredsstillte mine krav. Om dette nettstedet kan jeg si at SourceForge...

1. Er opprettet med et generelt formål
2. Bruker engelsk som arbeidsspråk
3. Tilbyr hosting av kode og web
4. Har alle de verktøy jeg krevde unntatt søkemulighet innenfor prosjektet. Kanskje kan man lage det selv.

²⁸ http://en.wikipedia.org/wiki/Comparison_of_open_source_software_hosting_facilities

²⁹ Listen over absolutte krav er gjengitt på side 33















³⁰ Listen over ønskelige egenskaper er gjengitt på side 33

³¹ Se en oversikt over andre egenskaper på side 33

5. Er gratis

SourceForge oppfyller også de ønskelige egenskapene meget bra³². Det kan i tillegg nevnes at det er mulig å utvikle applikasjonene hvor som helst, for så bare å distribuere gjennom SourceForge. Nettstedet er da også et nettsted bare for *nedlasting* av fri programvare.

Sourceforge.net angir selv sine tjenester slik³³:

Hosting	Development	Distribution	Community
Hosting is a pain; let us cater.	We've got software for your software.	Release for real – early and often.	We're your closest million friends.
 Code Hosting Code wants to be free. Host your code on our free and public, warm and cozy SVN, Git, Mercurial, Bazaar, or CVS servers.	 Tracker We all have issues. Track yours in our new and improved tracker.	 File Release Release your software for easy download on every platform you support. Our system will auto-detect your users' platform to deliver exactly what they need.	 Exposure Not THAT kind of exposure. Be seen by our millions of visitors, and on the first page of web search results. The world should know about your software.
 Web Hosting Help yourself to a cup of our Google juice. Host your project or personal website as a high-ranking SourceForge subdomain including shell access and web traffic analytics.	 Forums Hug it out. Collaborate with your community in our new and improved forums.	 Worldwide Mirror Network Put your software where the people are. Our download mirror network spans 5 continents so your users get the fastest download available.	 Millions of Users 2 Million heads are better than one. More users means more bug reports, more patches, and more feedback. Use our "Help Wanted" area to recruit talent for your project.
 Application Hosting Choose from a dozen available applications including Trac, MediaWiki, WordPress, and more. One click and we'll host any or all of them for you.	 Wiki We'll host a Trac or MediaWiki server for you.	 Statistics Track your download, tracker, forum, and code activity.	 Support Staff Our tireless support staff not only helps you during your development, but helps your users too. You need help, so get it here .
	 Blog If it's worth saying, say it right. We'll host a WordPress and/or Laconica server for you.		 Open Source Tradition Over the last 10 years we've helped launch prodigious projects like MySQL, Python, JRuby, JBoss, SugarCRM, and others.

Som det fremgår, er det en ganske omfattende liste. I tillegg er det verktøy som man kan benytte og som ikke står på listen her.

Mine erfaringer

Det kan være enkelt å velge Sourceforge.net bare fordi det er størst ("så mange brukere kan ikke ta feil"). Det følte jeg imidlertid som usikkert – jeg vet jo da ikke om dette nettstedet er det beste for mitt prosjekt. Jeg tok derfor en grundigere vurdering, gjengitt ovenfor. Det er arbeidskrevende fordi

- *det er mange slike FOSPHost nettsteder*
- *det er mange egenskaper å sjekke ut*

Allikevel vil jeg anbefale at man tar seg bryet – det vil være dumt å publisere på et nettsted der man ikke får noe resultat.

o. Publisering

Ovenfor har jeg sett på de viktigste rådene fra Fogel, og hva jeg gjorde i den forbindelse. I den grad det passet og jeg klarte det, har jeg fulgt rådene, som endte med at jeg valgte FOSPHost nettstedet *Sourceforge.net* for mitt prosjekt.

³² En oversikt er gjengitt på side 33

³³ <http://sourceforge.net/register/>

Det viste seg da at enkelt av de dokumentene jeg hadde klargjort, måtte gjøres litt om for å tilpasse prosjektet til nettstedets krav. Nettstedet hadde også spesielle krav som måtte oppfylles.

I dette kapitlet skal jeg beskrive hvordan publiseringen foregikk og hvilke ekstra krav som måtte tilfredsstilles.

Mitt prosjekt

For å publisere på Sourceforge.net må man ha en konto. Det er svært enkelt, og gratis, å sette opp.

Videre må man bekrefte at man vil følge "Terms of Use"³⁴. De er helt vanlige og ganske omfattende. Mest interessant er kanskje reglene om "Posting"³⁵ der man bekrefter

- eierskap til innholdet og at man ikke bryter med andres opphavsrettigheter
- at innholdet er lovlig

Sourceforge.net krever videre at du skal angi typen lisens for prosjektet ditt, og at den skal være

- compliant with the Open Source Initiative ("OSI")'s Open Source Definition (<http://www.opensource.org/docs/osd>) or
- certified as an "OSI-Approved License" (<http://opensource.org/licenses>).

GNU GPL, som jeg bruker, ligger greit innenfor disse kravene, og er godkjent av Opensource.org.

For å skape et prosjekt, må man oppgi et navn og foreslå navn på nettstedet. Jeg valgte å bruke navnet MultiStart og gi nettstedet samme navn (det fulle navnet blir <http://sourceforge.net/projects/multistart> da store bokstaver ikke tillates her). Man må også gi en beskrivelse av prosjektet på minst 50 tegn, maksimum 250. Den beskrivelsen jeg hadde laget var dessverre 1352 tegn, altså langt over det tillatte. Bare avsnitt to alene er altfor langt, nemlig 648 tegn. Jeg måtte altså lage en ny beskrivelse, og den måtte bli svært kort:

MultiStart starts several programs on a list with only one click, with a given interval and parameters. MultiStart helps you edit the list, and shows the list as buttons. Thus you can also use MultiStart as a menu and start the programs one by one.

Dette er 248 tegn, akkurat under det maksimalt tillatte.

Jeg mottok umiddelbart en velkomstmelding pr e-post³⁶, der det bekreftes at prosjektet er registrert og vil bli tilgjengelig om noen minutter. Faktisk tok det i mitt tilfelle bare noen sekunder.

Nettstedet inneholder da allerede følgende, generert automatisk³⁷:

- Mulighet for "rating" (den kan jeg skru av, men det ser jeg ingen grunn til)
- Diskusjonsforum "Open Discussion" med en første tråd med automatisk generert velkomstmelding fra meg
- "Bug Tracker"
- "Patches tracker"

³⁴ http://sourceforge.net/apps/trac/sitelegal/wiki/Terms_of_Use

³⁵ Reglene er gjengitt i sin helhet på side 35

³⁶ E-posten er gjengitt på side 35

³⁷ Full oversikt over informasjon og verktøy er gjengitt på side 35

- Prosjektinformasjon er registrert i Sourceforge.net ("sf-robot")

Det er fortsatt mye som kan gjøres:

1. Legge inn logo. Her lastet jeg den største (288x288 piksler) i png-format og det gikk helt greit.
2. Sette hjemmeside – jeg beholdt den som allerede er satt automatisk
3. Legg til "tags" (for søking) – det ventet jeg med
4. Lage en liste over "features" – det ventet jeg med
5. Under "Support" satte jeg "Forum: Help" som den foretrukne måten å få hjelp på.
6. Under "Develop" satte jeg inn skjermbilder. Det fremgår at formater mellom 150x150 og 640x480 piksler aksepteres i noen formater og begrenset til 300 Kb. Man kan ha maksimalt seks bilder. Mine to bilder var for store, og måtte derfor forminskes med en billedbehandler. Det ene ble dessverre ikke særlig klart, men jeg lot det være foreløpig. Det viste seg at bildene vises på nettstedet i alfabetisk rekkefølge, derfor kalte jeg det ene "A: Main view" og det andre "B: Edit view" for å få riktig rekkefølge ("1" og "2" hadde vel gjort samme nytten). Det viste seg også at "Thumb nail" uansett vises i liggende format – siden mitt bildet er i stående format, må jeg legge til endel på hver side av bildet, så formatet blir liggende.
7. I "Project Admin" bekreftet jeg at jeg ikke bruker kryptering.
8. Sourceforge har lister for "bugs" og "Feature requests" (man kan legge til flere, men det lar jeg være). Jeg rapporterte alt som er gjengitt ovenfor. Jeg la inn én og én, slik at jeg kunne sette kategori og prioritet for hver enkelt og evt. rapportere som løst etter hvert. Dette viser seg å være en relativt tidkrevende jobb. For hver endring fikk jeg dessuten en e-post om endringen (automatisk)
9. For å få bilde av meg selv i Sourceforge.net, måtte jeg skape en konto i gravatar.com. Deretter vistest bildet mitt i Sourceforge.net også.
10. Jeg lastet så opp programmet i kjørbær versjon, pakket med WinZip. Jeg vedla en "ReadMe"-fil som planlagt, men med mindre tilpasninger³⁸. Den la jeg også separat i mappen for "Executables (user files)" som jeg laget. Foreløpig merket jeg filen for "Windows".
11. Jeg gjorde tilsvarende i mappen "Source files (developers). Også her gjorde jeg mindre endringer i den planlagte teksten.
12. SourceForge gir automatisk en standard hjemmeside, men den er ikke særlig god. Man kan laste opp sin egen hjemmeside, som da erstatter den standardiserte. Det ønsket jeg. Det visste seg at jeg ikke kunne bruke det FTP-programmet (CuteFTP) som jeg vanligvis bruker til slikt. Jeg ble i stedet nødt til å laste ned og installere Filezilla³⁹ og da gikk det greit med anvisningene i SourceForge. Hjemmesidene laget jeg med Adobe DreamWeaver.

Mine erfaringer

Det var relativt enkelt – rent teknisk – å sette opp nettstedet, når alt innholdet først var klart. Det var imidlertid ganske tidkrevende. Det medførte bl.a. at jeg måtte inn på andre nettsteder også (gravatar.com) og at jeg måtte installere et nytt program (Filezilla). Bildene jeg skulle bruke måtte endres i en billedbehandler, og tekstene måtte tilpasses noe.

³⁸ Se den rettede filen på side 35

³⁹ Den kan lastes ned fra <http://filezilla-project.org/>

5. Konklusjon og diskusjon

Det er økende bruk og publisering av Fri programvare. EU mener også at det er samfunnsmessig ønskelig. Høyskolen jeg arbeider ved, har dette som ett av sine satsningsområder og min fagseksjon interesserer seg også for dette. Det har derfor vært interessant å forsøke hva slik publisering innebærer av arbeid, dvs. hvilke arbeidsmomenter som inngår, hva som er vanskelig/enkelt og hva det krever av tid og andre ressurser.

Som jeg nevnte under en kort beskrivelse av forskningsdesign ovenfor, tenker jeg på denne undersøkelsen mest som en case-studie der jeg selv er svært aktiv part. Svært meget av det som er beskrevet er følgelig sterkt subjektivt. Jeg mener det er tre sentrale elementer i denne rapporten:

1. Hva Karl Fogel anbefaler (normativt og generelt)
2. Hva jeg gjorde (referat av mine aktiviteter)
3. Hva jeg erfarte (deskriptivt og subjektivt)

Hvilke av disse tre delene som er nyttig for den enkelte leser, kan sikkert variere.

Om Fogels råd

Karl Fogel har solid grunnlag for sine råd, basert på hans egen, lange erfaring med utvikling av fri programvare og deltakelse i store, slike prosjekter. De rådene som jeg har gjengitt ovenfor, er hentet fra hans bok⁴⁰ om emnet, og leseren kan hente den på Internett og lese den selv. Jeg kan varmt anbefale denne boken.

Jeg opplevde selv Fogels råd som relevante, lette å forstå og enkle å følge. Ingen av dem skapte noen problemer for meg. De viktigste vanskelighetene som jeg hadde, skyldtes enten at jeg hadde gjort for meget med programmet før jeg begynte å følge rådene, og at nettstedet jeg valgte hadde visse begrensninger som jeg ikke kjente til under arbeidet.

Om mitt arbeid

Jeg har beskrevet hvilket arbeid jeg gjennomførte for hvert råd. Det er beskrevet i tilknytning til hvert råd av hensyn til leseren, selv om det faktisk ikke alltid ble gjort slik. Noe arbeid ble gjort samtidig (men er henført ovenfor til hvert sitt råd) og noe parallelt. Jeg tror allikevel ikke at dette gir noe falsk bilde.

Man bør være oppmerksom på at noe av mitt arbeid ble gjort i en rekkefølge som jeg ikke anbefaler. Det tar jeg opp i neste avsnitt.

"Prosjektsidene" for mitt prosjekt er på <http://sourceforge.net/projects/multistart/>. Hjemmesiden er <http://multistart.sourceforge.net/>⁴¹.

Om mine erfaringer

Også mine erfaringer er knyttet til hvert sitt råd. Det kan kanskje være interessant med en oppsummering av dem.

Fogel anbefaler at man først sjekker om andre har gjort det samme. Jeg gjorde det, men fant først ingen andre like. Det var derfor jeg laget programmet i sin tid. Senere sjekket jeg grundigere, og fant da flere som liknet svært. Dette er følgelig en noe som bør kontrolleres

⁴⁰ Som nevnt ovenfor, finnes boken i sin helhet på Internett (<http://producingoss.com/>).

⁴¹ Førstesiden av "Prosjektsidene" og hjemmeområdet er gjengitt på side 39.

Distribusjon og publisering av fri programvare

tidlig i arbeidet. Det kan jo – i tråd med hele idéen om fri programvare – være mye bedre å delta i et eksisterende prosjekt enn å starte et nytt.

Hvis man da beslutter å starte et eget prosjekt, bør man tidlig undersøke mulige navn. Det er mye arbeid å endre navn etter at programmeringen er godt i gang. Man bør da samtidig sikre seg de nødvendige Internett-domener.

Deretter bør man bestemme seg for et nettsted for publisering. Jeg støtter Fogel i at det er fornuftig å bruke et nettsted som er dedikert for frie programvare prosjekter, og kan anbefale *SourceForge.net*. Hvis man velger nettsted tidlig, unngår man å måtte endre dokumentasjon o.l. for å tilpasse seg nettstedets krav. Isteden kan man lage dem overensstemmende med kravene med en gang. Det er ganske mye arbeid med "Mission Statement", brukerdokumentasjon, utviklingsstatus og utviklerdokumentasjon – mer enn jeg hadde tenkt på forhånd – og det er ergerlig å lage slik dokumentasjon som ikke godtas av nettstedet.

Noen oppgaver gikk ganske greit og tok ikke særlig lang tid:

- Å lage logo/ikon
- Å velge lisens (man bør ikke lage sin egen)
- Å lage beskrivelse til utviklere
- Å lage bilder av skjemaene

Andre oppgaver var mer tidkrevende, selv om de kanskje ikke var så vanskelige:

- Å lage fornuftige kommentarer i kildekoden
- Det var enkelt å sette opp nettstedet på *SourceForge.net*, men ganske tidkrevende (det kan nok variere fra nettsted til nettsted)
- Å lage brukerhjelp, selv om den var svært enkel

Det var både vanskelig og tidkrevende å lage "Mission Statement".

Fra andre prosjekter vet jeg at det kreves mye arbeid å lage god brukerhjelp, men det har jeg ikke gjort denne gang.

Endelig konklusjon

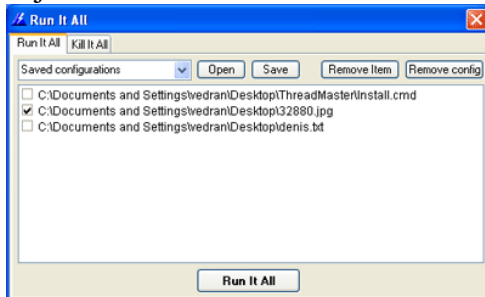
Jeg anbefaler andre som er tenker å lage et fritt programvareprosjekt å starte med en grundig kontroll av om det allerede finnes et liknende, fritt program. Hvis et slikt finnes, er det bedre å "henge seg på" det, istedenfor å lage sitt eget. Hvis man etter denne kontrollen fortsatt vil lage sitt eget prosjekt, bør man velge et FOSPHost nettsted for publisering. Da får man mange, nyttige verktøy inkludert. Deretter kan jeg anbefale at man henter Karl Fogels råd for slike prosjekter (<http://producingoss.com>) og følger dem.

Det er relativt tidkrevende, men ikke spesielt vanskelig, å komme fra et program som man selv har og er fornøyd med, til det er ferdig publisert som et fritt programvareprosjekt.

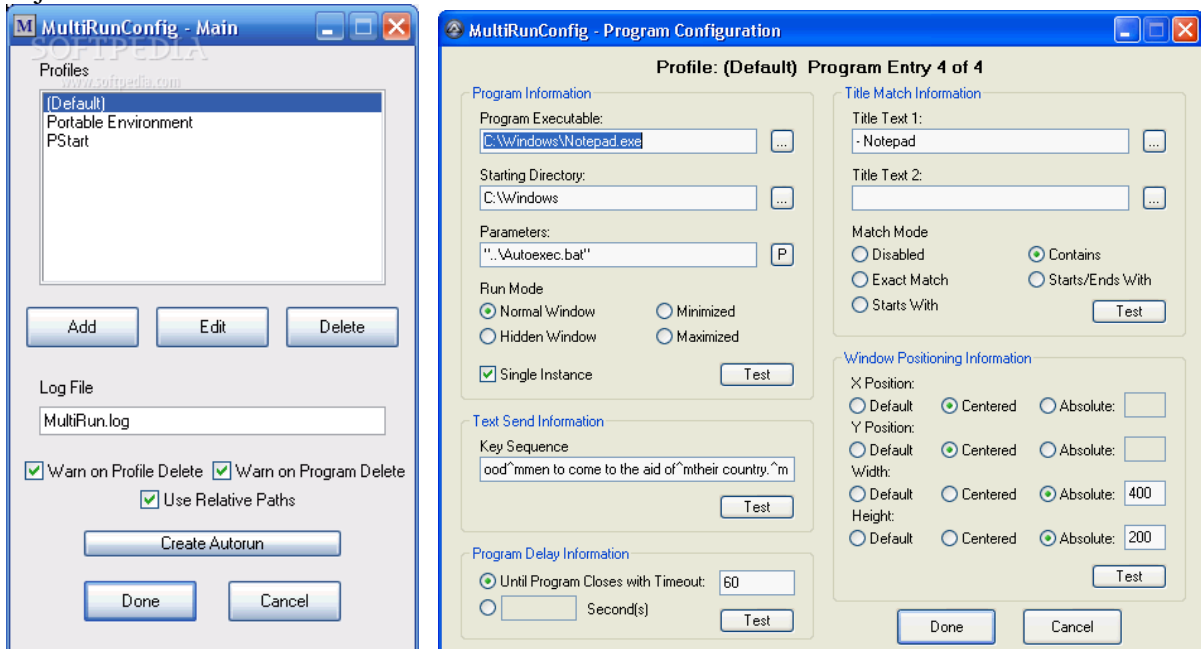
6. Appendiks – Detaljer til enkelte kapitler

Ad Fogel: Se deg rundt

Skjerm bilde for *Run It All*:



Skjerm bilder for *MultiRun*⁴²:



MultiRun beskriver slik:

MultiRun is an extremely flexible multi-program launcher. In addition to launching a series of programs in sequence, the program will also let you control command line parameters, the window size and position, you can send keystrokes to the program, and you can wait for programs to close or a fixed period of time between program launches.

MultiRun is completely "portable" and ready for your flash drive. There is no dependency on the system registry, dll libraries, ocx controls, or anything else. MultiRun can also be used as a startup control for your flash drive giving you choices as to the environment that you want to bring up when you plug your flash drive into a computer.

⁴² Det er programmert med *AutoIt* – et gratis (ikke fri programvare) Basic-liknende skriptspråk for automatisering av Windows GUI. *MultiRun* er forøvrig også en Java-klasse – en del av JAS biblioteket for simuleringsapplikasjoner (<http://jaslibrary.sourceforge.net/>).

Ad Fogel: Velg et godt navn

Ikonet/logoen for mitt program ser slik ut:



Meningen er å illustrere en liste med programmer som lastes inn i PC'en.

Ad Fogel: Lag en klar "Mission statement"

Min "Mission Statement":

***The problem:** When you start your PC, you seem to wait forever. There are so many programs that are started. Some of these are started because you want them to – by placing a link to them in the Startup folder on the Start menu. But sometimes you just want to load only a few essential programs, at other times you want the whole package. Such conditions cannot be specified in the Startup folder.*

***The solution:** Place the programs that you sometimes want to start with the PC and sometimes not, in a list. **MultiStart** will then start these programs with a single click – if you want to – or leave them unstarted if that's what you prefer this time. **MultiStart** will help you edit the list and will show them as buttons on its form. You can click once to start many, or click one button at a time to start only the one associated with that button. If you want to, you can add parameters, so that for instance your net browser is started with a specific web page. You can specify an interval between each program start (actually this can be quicker).*

***The license:** **MultiStart** is open source and absolutely free. It is guaranteed to remain free through its GNU GPLv3 (<http://www.gnu.org/licenses/gpl.html>) license. This means that you can download the complete source code (Java) and change it and distribute it as much as you like (providing only that you yourself do so with the same license).*

Ad Fogel: Lag god liste over funksjonalitet og krav

Min liste over funksjonalitet og krav, ble slik:

Features:

- *Shows two groups of buttons. Each button is associated with a program (with parameters) so that a click on the button starts the associated program. The button icon is automatically fetched from the program file.*
- *By a single click, all programs associated with the buttons in one of the groups, can be started.*
- *A delay can be set between each such start.*
- *Whilst the programs are being started, MultiStart can stay on top or minimize itself.*
- *These settings (delay and on top or minimized) can be saved to an initialization file, which can also be edited with a standard text editor.*
- *The buttons shown are based on a list in plain text. This can be edited with any text editor. There is also a special window (frame) for editing this file graphically.*

Distribusjon og publisering av fri programvare

Requirements:

- *MultiStart has been written totally in Java and should run on virtually any operating system. It has, however, so far only been tested on Windows.*
- *Java Run Time System (<http://www.java.com/en/download/manual.jsp>) must be installed. (Most machines have this already – and it's free. You can have Sun check this (<http://www.java.com/en/download/installed.jsp?detect=jre&try=1>) for you on-line.)*
- *MultiStart takes up some 170 KB of your hard disk (and your RAM when it's running).*

License:

MultiStart is licensed under GNU GPLv3 (<http://www.gnu.org/licenses/gpl.html>) – free and open source software.

Ad Fogel: Angi utviklingsstatus

Her er følgelig (foreløpige) min liste:

Known bugs:

- *None*

Known issues:

- *MultisStart has only been tested in Windows Vista. It should be tested on other operating systems, especially on Linux.*
- *When MultiStart has been minimized and has finished starting a group of programs, it should revert to top window state (on top of all other windows). This is not working – it is maximized, but not on top.*
- *MultiStart main frame does not have the right size.*

Known missing features:

- *The user can select whether a particular program should be started in minimized, normal or maximized state, but this is not working.*
- *When the user has changed which buttons to show, the main frame is not updated on return.*

Enhancements (unordered):

- *Better user manual and installation instructions.*
- *Allow program to reside – if user so wishes – in Notification Area (down to the right of Windows screen).*
- *Show own logo on all frames, program and short cuts (the logo has been made)*
- *Drag-drop of new programs should be allowed.*
- *The program has been made with NetBeans IDE. This means that some control has been lost. Reprogram without NetBeans.*
- *The button icon is now automatically set to the first of possibly several icons saved in the program icon. Let the user select one (if there is more than one).*
- *The pause between start of several programs is programmed in a way that “steals” processor cycles. This should be reprogrammed.*
- *Show splash frame at start, but allow user to turn it off (choice to be saved in ini-file).*
- *Allow user to save several, different “lists” of programs for top frame (the one used for launching several programs with time delay if required).*

Ad Fogel: Brukerdokumentasjon

Min omtale av installasjon av litt om bruk:

Installation on a Windows PC:

1. *Download the zip-file (right click on link and choose "Download")*
2. *Unzip the file to the directory of your choice*
3. *That's it, you're done*
4. *On first start, the application may not find certain files. Don't worry, they will be made automatically.*
5. *You are now ready to specify which programs you want to launch with MultiStart.*

Some hints:

6. *I prefer to start programs from the desktop. To make a short cut on your desktop, right click on the file "MultiStart.jar", and select "Send To...Desktop". You may then drag this short cut – if you prefer – to your start menu or to the "Quick Launch" area of your task bar (in the lower left of your desktop) or both.*
7. *Choose how the application should behave and remember to save these choices. Feel free to experiment, they can be changed and saved at any time.*
8. *In the top frame (actually called a panel) you place those programs that you will be able to start with one click. In the bottom you can place programs that you start often, and would like to have handy.*
9. *You should add the short cut to the folder "Startup" in the "All programs" part of the "Start menu". This will launch MultiStart automatically when you star or restart your machine.*

Ad Fogel: Utviklerdokumentasjon

Filen som angir hvilke programmer som skal startes ser slik ut:

```
// The format is <panel>,<program name with full path>,<argument>,  
// <shortname shown to user>,<framestyle (not currently in use)>.  
  
// Panel = 1 means upper panel, panel = 2 means lower panel.  
// All arguments are compulsory.  
// Lines starting with "//" and empty lines are ignored.  
  
1,C:\Program Files\Taskbar Shuffle\taskbarshuffle.exe,,TaskbarShuffle,MIN  
1,C:\PMAIL\Programs\winpm-32.exe, -A -I knuth,Pegasus Mail (<default>),MIN  
1,C:\Program Files\Clock\clock.exe,,Clock!,NORMAL  
2,C:\Program Files\UsbBoost\TurboHddUsb.exe,,USB Boost LaCie,MIN  
// 2,C:\Program Files\OpenVPN\bin\openvpn-gui.exe,,VPN HiBu,NORMAL  
// 2,C:\Program Files\Copernic Desktop Search  
2,C:\Program Files\SyncToy 2.1\SyncToy.exe,,SyncToy,NORMAL  
2,C:\Program Files\WinZip\WINZIP32.EXE,,WinZip,MAX  
...
```

Ini-filen som leses ved oppstart og setter noen verdier, ser slik ut:

```
// Inifile.  
// First value below = 'Always on top (Minimize is always the opposite)'  
// Second value below = 'Pause (sec) between each'  
/////////////////  
false  
1
```

Distribusjon og publisering av fri programvare

Her er et eksempel på kommentar til prosedyren `createButtons`:

```
/**
 * Create buttons from the lines in <code>buttonList</code>.
 * Only lines (if any) conforming to the syntax for buttons are used.
 * The result is placed in <code>buttonVector</code>.<br>
 * <strong>Note:</strong> <code>MouseListener</code> is <i>not</i>
added.
 * (The listener changes values in the <code>MainView</code> and should
therefore
 * be added there.)
 * @author Knut W. Hansson
 * @version 2010.04.26
 */
public static void createButtons() throws Exception{
```

Dette gir følgende dokumentasjon med JavaDoc:

Method Detail

createButtons

```
public static void createButtons()
    throws java.lang.Exception
```

Create buttons from the lines in `buttonList`. Only lines (if any) conforming to the syntax for buttons are used. The result is placed in `buttonVector`.

Note: `MouseListener` is *not* added. (The listener changes values in the `MainView` and should therefore be added there.)

Throws:

`java.lang.Exception`

Koden er f.eks. kommentert slik:

```
try {
    if (!hasButtons()){
        throw new Exception ("List has no buttons");
    }
    else {
        // Convert lines to buttons and add to vector
        for (String line:buttonList){
            if (!(line == null||line.startsWith("//")||line.isEmpty())){
                // Create buttonobject and set fields
                // NB! No mouselistener is added
                try {
                    ButtonObject newButton = new ButtonObject(line);
                    buttonVector.add(newButton); // add button to vector
                }
                catch (Exception ex){
```

De fete linjene er kommentarer. Her har jeg satt dem med fet skrift, men i en Java-editor vil de gjerne fremstå med egen farge e.l. så de er lette å skille fra selve koden.

Beskrivelsen for utviklere ser slik ut:

For developers:

This application has been programmed in Java. The IDE NetBeans⁴³ has been used. It will be difficult to add or change code without it (see wanted enhancements).

⁴³ <http://netbeans.org/>

Distribusjon og publisering av fri programvare

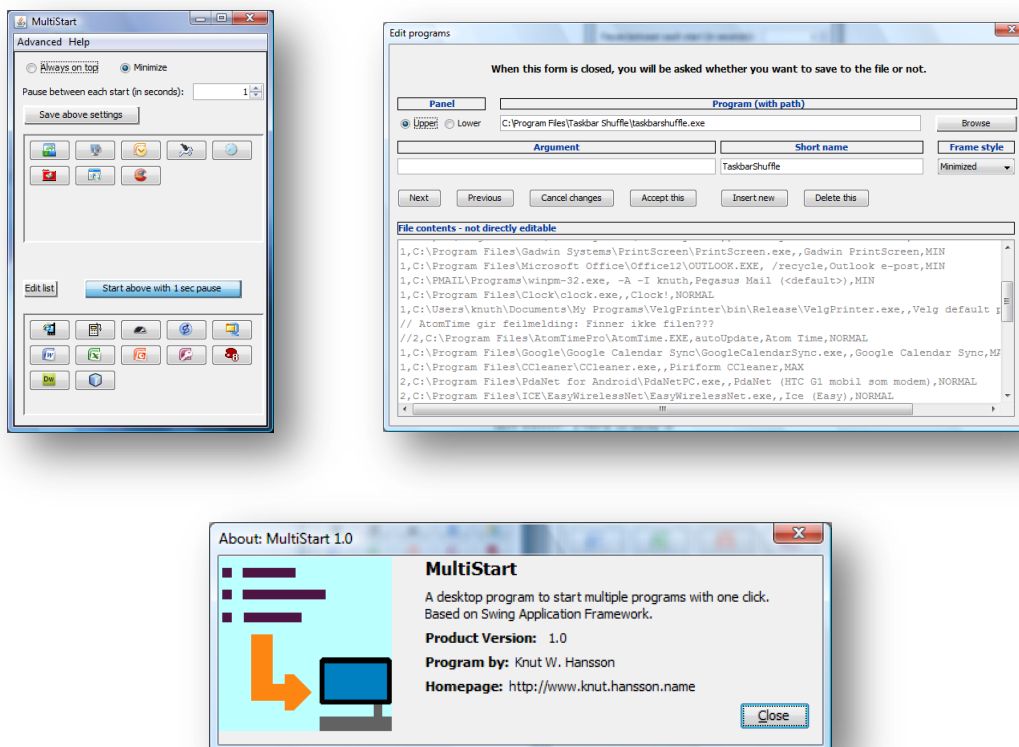
Two text files are used, one for initiation (“MultiStart.ini”) and one for the wanted buttons (“MultiStartButtons.txt”). The application reads and writes to both of them.

The code is commented and documented. Please keep the documentation up to date with changes. The preferred language is English.

Suggested changes should be submitted through this project site, and will be reviewed by me for the time being. Suggestions that are accepted will be published here as a new (sub) version. If you would like to take over the project responsibilities, please feel free to suggest yourself to me.

Ad Fogel: Eksempler på output og skjermbilder

Passende bilder av min applikasjon kan se slik ut (sterkt forminsknet her):



Ad Fogel: Bruk nettsted for fri programvare prosjekter

Under grovsilingen så jeg bort fra nettsteder som ikke oppfylte disse kravene:

1. Nettsteder som er opprettet for spesielle formål forskjellig fra mitt. Eksempler er *freedesktop.org* som er for Linux infrastruktur, *MacOS Forge* for Mac OS X og *mozdev* for Mozilla-prosjekter.
2. Nettstedet må bruke engelsk språk, det gjelder f.eks. ikke *ourproject* som er mye på spansk.
3. Nettstedet må tilby hosting (lagring) av kildekode og web-hosting (lagring av eget nettsted for prosjektet). F.eks. gjør hverken *CodePlex* (Microsoft) eller *BerliOS* det.
4. Nettstedet må tilby verktøy som jeg mener er vesentlige. Det vil si i alle fall følgende verktøy:
 - a. Wiki for definisjoner, tips o.l.
 - b. Mailinglist for å øke kommunikasjonen mellom alle interessenter
 - c. Forum

Distribusjon og publisering av fri programvare

- d. Annonsering, dvs. nyheter
 - e. Nedlasting av seneste versjon (release)
 - f. Statistikk for prosjektet
 - g. Backup. Jeg regner naturligvis med at alle FOSPHost nettsteder ordner selv for hele nettstedet, men det kan være greit å kunne ta egne backups og ikke minst restores for bare mitt prosjekt.
 - h. Gode søkemuligheter for hele nettstedet og helst også egen søkefunksjon innenfor prosjektet.
5. Det bør være gratis eller svært billig å ha sitt prosjekt på nettstedet.

Andre egenskaper ved nettstedene som er **ønskelige**, er:

1. FOSPHost nettsteder drives enten av en større sponsor, men ofte finansieres de med reklame. Reklamen er naturligvis en ulempe, men jeg oppfatter den ikke som kritisk.
2. Det er stor forskjell på hvor mange brukere og prosjekter nettstedene har. Det er en fordel med mange brukere, fordi sjansen da er større for at noen av oppdager mitt prosjekt og fatter interesse for det. Fordelen ved å ligge på et stort nettsted med mange prosjekter, er både at sjansen for at nettstedet nedlegges er mindre, og at mange som er interessert i fri programvare trekkes dit for å lete etter interessante programmer og prosjekter. Tilsvarende kan antall bug reports gi en indikasjon på et aktivt nettsted.
3. Eieren av nettstedet bør være solid. Dette har igjen betydning for nettstedets overlevelse.
4. Nettstedet bør ha eksistert noen år. Det indikerer en viss stabilitet.
5. Det kan være en fordel med et nettsted med stor trafikk, fordi det gir en indikasjon på hvor kjent nettstedet er og/eller hvor ofte nettstedet kommer høyt opp ved et søk.
6. Nettstedene har som regel verktøy for versjonskontroll, og alle jeg vurderte har det. Verktøyene gjør det enkelt å legge ut nye versjoner og får dem merket med endringer og en forklarende tekst om hva som er endret. De virker nok på forskjellig måte (jeg har ikke prøvd dem), men jeg har ingen spesiell preferanse. De bør imidlertid ha ett slikt verktøy, i tilfelle jeg ønsker å implementere det. Kjente verktøy for versjonskontroll er bl.a. Bazar, CVS, Git, Mercurial og Subversion. Videre er Microsoft TFS noe mer omfattende (inkluderer tracking av kildekoden også). En omfattende sammenlikning av versjonskontroller finnes på Wikipedia⁴⁴.

Videre har nettstedene gjerne egenskaper som **ikke betød noe** for meg:

1. Noen FOSPHost nettsteder utelukker posting fra enkelte land (som f.eks. Cuba, Libya, Nord Korea, Syria), men det betyr lite for meg.
2. Nettstedene tilbyr mange andre verktøy enn de som er nevnt ovenfor. Disse verktøyene er ikke kritiske for mitt prosjekt, synes jeg:
 - a. Shell server⁴⁵: Logg inn og få kommandoorientert grensesnitt til manipulasjon av innhold.
 - b. NNTP⁴⁶: UseNet tjenester (protokoll for overføring av UseNet).
 - c. Mulighetene for ”private” samtaler/forums (der bare noen få kan delta).
 - d. Build system (kompilering)
 - e. Team (noen skiller seg ut i egen gruppe for å løse bestemte oppgaver)
 - f. Jeg har ikke tenkt å bruke blogg – jeg anvender isteden forum.

⁴⁴ http://en.wikipedia.org/wiki/Comparison_of_revision_control_software (nov. 2010)

⁴⁵ <http://sourceforge.net/apps/trac/sourceforge/wiki/Shell%20service>

⁴⁶ <http://no.wikipedia.org/wiki/NNTP>

Distribusjon og publisering av fri programvare

- g. Bounties er et verktøy som gjør det enkelt å be om bidrag mot betaling. Beste, eller første aksepterte, får en pris. Jeg har ikke tenkt å be om bidrag på denne måten, så dette er irrelevant for meg.
 - h. Arch Linux er et verktøy som gjør det enkelt for utviklere å pakke og distribuere programvare for Linux. Det er ikke særlig aktuelt for meg.
3. Nettstedet drives jo også med programvare. For noen forkjempere av fri programvare, kan det være et poeng at nettstedet selv bruker fri programvare. Noen av dem kan også ha sine egne systemer som fritt prosjekt så man kan delta i utviklingen av selve nettstedet. Jeg legger ikke selv særlig vekt på dette.

Når det gjelder de *ønskelige* egenskapene, så kan det nevnes at SourceForge...

1. drives av et større kommersielt selskap (Geeknet.inc).
2. har flest brukere og prosjekter av alle FOSPHost nettsteder – pr februar 2009 var det over 230.000 prosjekter og mer enn 2 millioner brukere registrert.
3. har eier som oppfattes som solid
4. har holdt på siden 1999 – det må anses som lenge i denne sammenheng (de første kom i 1997 og de fleste er etablert på 2000-tallet)
5. har stor trafikk. Alexa måler internett-trafikken til nettsteder⁴⁷. Globalt ligger SourceForge på ca 136. plass (varierer fra 100 til 200 over tid). I kategorien Open Source Project Hosting, kommer SourceForge på topp våren 2010), foran GitHub som jeg har ekskludert av andre grunner. Hver dag er rundt 0,7 % av alle brukere innom SourceForge og ser da på ca 3 nettsider.
6. har flere verktøy for versjonskontroll

Ad Publisering

Regler for nettstedet, som må aksepteres:

When you create or make available Your Content, you represent and warrant that:

- a. you own or have sufficient rights to post or make available Your Content on or through SourceForge.net;
- b. the posting or making available of Your Content on or through SourceForge.net does not violate the privacy rights, publicity rights, copyrights, contract rights or any other rights, including intellectual property rights, of any person or entity;
- c. you have fully complied with any third-party licenses relating to Your Content, agree to pay for all royalties, fees and any other monies owing any person by reason of any of Your Content that you posted on or through SourceForge.net;
- d. Your Content does not contain any viruses, worms, Trojan horses, malicious code or other harmful or destructive content;
- e. Your Content is not obscene, lewd, lascivious, excessively violent, harassing, libelous or slanderous, does not advocate the violent overthrow of the government of the United States, does not incite, encourage or threaten immediate physical harm against another;
- f. Your Content does not endorse or promote racism, bigotry, hatred, or physical harm of any kind against another group or individual, and does not discriminate, incite harassment or advocate harassment of any group or individual;
- g. Your Content does not contain material that solicits personal information from anyone under 18 or exploit people under the age of 18 in a sexual or violent manner, and does not violate any federal or state law concerning child pornography or otherwise intended to protect the health and well-being of minors;
- h. if your employer has rights to intellectual property you create, you have either
 - i. received permission from your employer to make available Your Content, or
 - ii. secured from your employer a waiver as to all rights in or to Your Content;
- i. Your Content does not constitute, contain, install or attempt to install or promote spyware or malware or any other computer code (whether on Geeknet's or others computers or equipment) intended to, or that does, enable you or others to gather information about or monitor the online or other activities of another party, unless it discloses such functionality to the user in a clear and conspicuous manner and the user affirmatively consents to such use; and,

⁴⁷ <http://www.alexacom/topsites/global>

Distribusjon og publisering av fri programvare

Your Content does not otherwise violate, or link to material that violates, any provision of these Terms or any local, state, or federal law or regulation, including any law designed to regulate electronic advertising

Velkomst-e-posten fra nettstedet:

Welcome to SourceForge.net!

The following project has been registered for hosting:

Descriptive Name: MultiStart
Unix Name: multistart
URL: <https://sourceforge.net/projects/multistart>

By using the SourceForge.net site, you agree to be bound by the terms and conditions of the SourceForge.net Terms of Use Agreement.

SourceForge.net project hosting is available solely to Open Source software development projects. If you are not releasing your project under an Open Source license, please contact SourceForge.net staff for assistance (see "Support", below).

Project Administration:

To manage your project, login and visit
https://sourceforge.net/project/admin/?group_id=321705

Getting Started:

Review our getting started guide:
<http://p.sf.net/sourceforge/getstarted>

Select an Open Source license and update the Trove categorization for this project:

https://sourceforge.net/project/admin/group_trove.php?group_id=321705

Enable the project features you want to use:
https://sourceforge.net/project/admin/features.php?group_id=321705

Support:

SourceForge.net maintains a full-time support team; we are ready to assist you with any questions you have about your new project.


We can be contacted by submitting a ticket at:
<http://p.sf.net/sourceforge/submitticket>



We may also be reached directly during business hours via IRC:
<http://p.sf.net/sourceforge/irc>


-- the SourceForge.net crew


Distribusjon og publisering av fri programvare


Informasjon og verktøy som nettstedet genererte automatisk og umiddelbart ved opprettelse:


Ratings and Reviews (Disable Reviews) 
Be the first to post a text review of MultiStart. Rate and review a project by clicking thumbs up or thumbs down in the right column.

Project Feed Show: 
 **Forum thread added**
knuthansson created the [Welcome to Open Discussion](#) forum thread
posted by knuthansson 2 minutes ago

 **Forum added**
knuthansson created the [Open Discussion](#) forum
posted by knuthansson 2 minutes ago

 **Tracker added**
knuthansson created the [Bugs](#) tracker
posted by knuthansson 2 minutes ago

 **Tracker added**
knuthansson created the [Patches](#) tracker
posted by knuthansson 2 minutes ago

 **Project Information Updated**
sf-robot registered the [MultiStart](#) project
posted by sf-robot 2 minutes ago

The following features are currently enabled:

Enabled	Feature	Category	Status	Options
<input checked="" type="checkbox"/>	Backups	Data Recovery	Always On	XML Export , Hosted Apps
<input checked="" type="checkbox"/>	File Manager	Download	Always On	Manage
<input checked="" type="checkbox"/>	Help Wanted	Recruiting	Always On	Submit , Manage
<input checked="" type="checkbox"/>	Project Database (MySQL)	Web Hosting	Always On	Manage
<input checked="" type="checkbox"/>	Project Web	Web Hosting	Always On	
<input checked="" type="checkbox"/>	Virtual Hosts (VHOSTs)	Web Hosting	Always On	Manage
<input checked="" type="checkbox"/>	Forums	Forums	Enabled	Manage
<input checked="" type="checkbox"/>	Mailing Lists	E-Mail	Enabled	Manage
<input checked="" type="checkbox"/>	Project News	News / Blog	Enabled	Submit , Manage
<input checked="" type="checkbox"/>	Screenshots	Media	Enabled	Manage
<input checked="" type="checkbox"/>	Subversion	Source Control	Enabled	Manage
<input checked="" type="checkbox"/>	Tracker	Bug / Defects	Enabled	Manage

Distribusjon og publisering av fri programvare

SourceForge.net > Find Software > MultiStart

EDIT



MultiStart by knuthansson

EDIT

Summary | Files | Support | Develop | Tracker | Mailing Lists | Forums | Code | Project Admin

MultiStart starts several programs on a list with only one click, with a given interval and parameters. EDIT
MultiStart helps you edit the list, and shows the list as buttons. Thus you can also use MultiStart as a menu and start the programs one by one.

View all files >

EDIT

<http://multistart.sourceforge.net>

EDIT

TAGS

EDIT

Features:

- Add a new feature
- Add a new feature
- Add a new feature
- Add a new feature
- Add a new feature

save changes

Registered: 2010-05-10

Filen ReadMe.txt for brukere, slik den ble til slutt:

Installation on a Windows PC:

1. Download the zip-file
2. Unzip the file to the directory of your choice
3. That's it, you're done
4. On first start, the application may not find certain files.
Don't worry, they will be made automatically.
5. You are now ready to specify which programs you want to launch with MultiStart.

Some hints:

6. I prefer to start programs from the desktop.
To make a short cut on your desktop, right click on the file "MultiStart.jar", and select "Send To...Desktop". You may then drag this short cut - if you prefer - to your start menu or to the "Quick Launch" area of your task bar (in the lower left of your desktop) or both.
7. Choose how the application should behave and remember to save these choices.
Feel free to experiment, they can be changed and saved at any time.
8. In the top frame (actually called a panel) you place those programs that you will be able to start with one click. In the bottom you can place programs that you start often, and would like to have handy.
9. You should add the short cut to the folder "Startup" in the "All programs" part of the "Start menu". This will launch MultiStart automatically when you start or restart your machine.

THIS FILE IS INCLUDED IN THE DISTRIBUTION ZIP-FILE

Distribusjon og publisering av fri programvare

ReadMe-filen for utviklere, slik den ble til slutt:

MultiStart - For developers:

This application has been programmed in Java. The IDE NetBeans (<http://netbeans.org/>) has been used. It will be difficult to add or change code without it.

Two text files are used, one for initiation ("MultiStart.ini") and one for the wanted buttons ("MultiStartButtons.txt"). The application reads and writes to both of them.

The code is commented and documented. Please keep the documentation up to date with changes.
The preferred language is English.

Suggested changes should be submitted through this project site, and will be reviewed by me for the time being. Suggestions that are accepted will be published here as a new (sub) version. If you would like to take over the project responsibilities, please feel free to suggest yourself to me.

~Knut W. Hansson

Ad Konklusjon og diskusjon

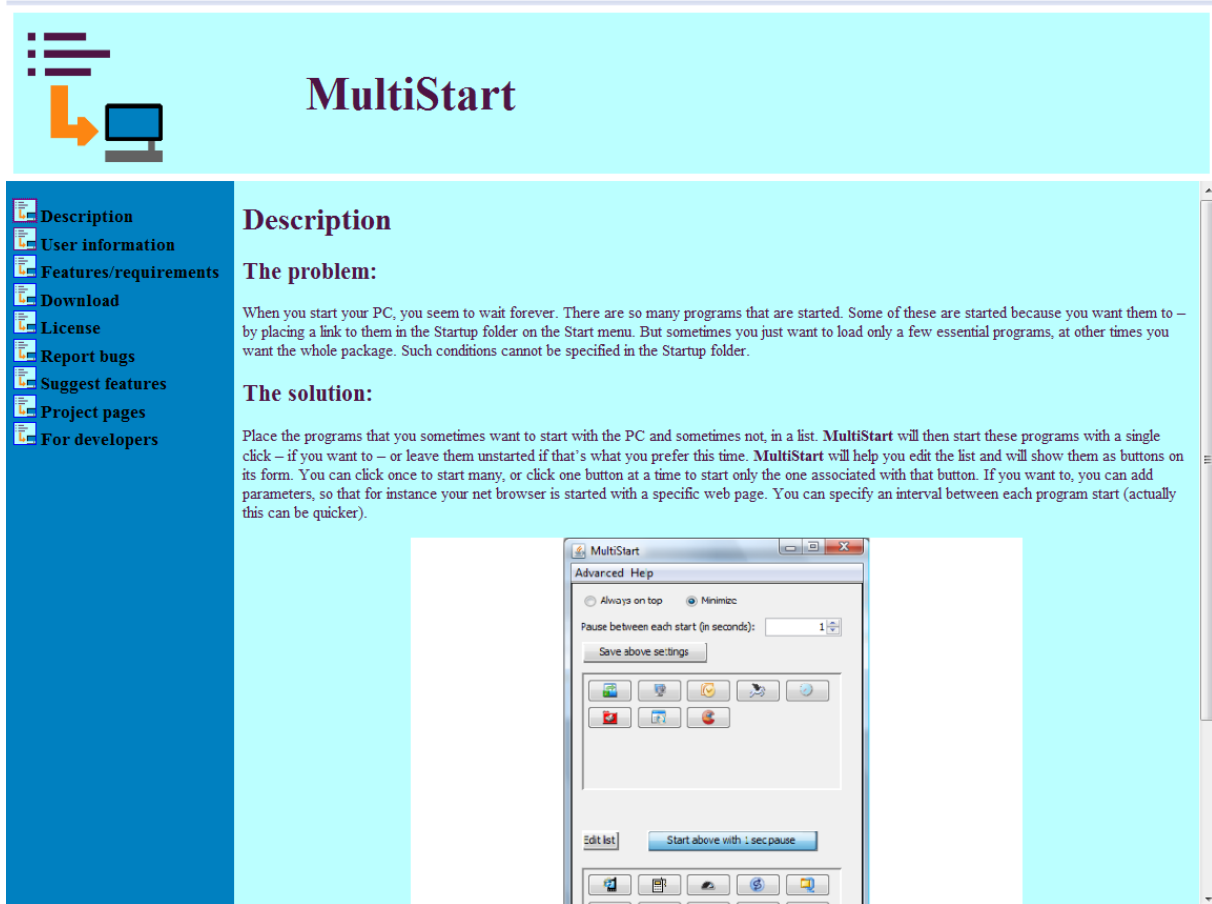
Førstesiden av "Prosjektsidene" ser slik ut:

The screenshot shows the SourceForge project page for MultiStart. At the top, there's a navigation bar with 'sourceforge' logo and 'FIND AND DEVELOP OPEN SOURCE SOFTWARE'. Below that, there are links for 'Find Software', 'Develop', 'Create Project', 'Blog', 'Site Support', and 'About'. A search bar is on the right. The main content area shows the project name 'MultiStart by knuthansson' with tabs for 'Summary', 'Files', 'Support', and 'Develop'. A description of the project is provided, followed by a 'Download Now!' button and a link to 'View all files'. Below this is a 'Project Feed' section showing recent file releases. On the right side, there are 'Ads by Google' for Groupon Shopping and a 'Rate and Review' section.

Legg merke til reklamen, som ikke er til å unngå. Det er prisen for gratis nettplass.

Distribusjon og publisering av fri programvare

Førstesiden av prosjektets hjemmeområde ser slik ut:



MultiStart

- Description
- User information
- Features/requirements
- Download
- License
- Report bugs
- Suggest features
- Project pages
- For developers

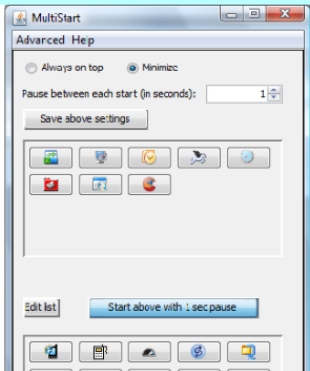
Description

The problem:

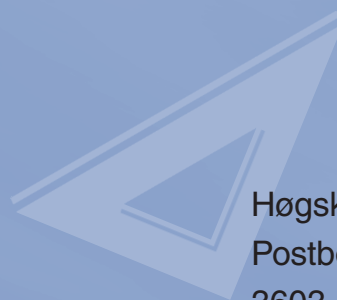
When you start your PC, you seem to wait forever. There are so many programs that are started. Some of these are started because you want them to – by placing a link to them in the Startup folder on the Start menu. But sometimes you just want to load only a few essential programs, at other times you want the whole package. Such conditions cannot be specified in the Startup folder.

The solution:

Place the programs that you sometimes want to start with the PC and sometimes not, in a list. **MultiStart** will then start these programs with a single click – if you want to – or leave them unstarted if that's what you prefer this time. **MultiStart** will help you edit the list and will show them as buttons on its form. You can click once to start many, or click one button at a time to start only the one associated with that button. If you want to, you can add parameters, so that for instance your net browser is started with a specific web page. You can specify an interval between each program start (actually this can be quicker).

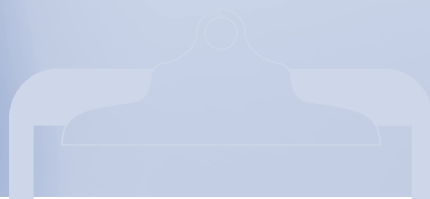


Her er det ingen reklame, selv om også disse sidene er lagret hos SourcForge.net.



Høgskolen i Buskerud
Postboks 235
3603 Kongsberg
Telefon: 32 86 95 00
Telefaks: 32 86 98 83
www.hibu.no

ISSN 0807-4488



HØGSKOLEN
i Buskerud