

# HTML Dokumentasjon av ActionScript 3.0 kode

## Introduksjon

I dokumentet UML anvendt i IT-2 (<http://tip.no/skole/pdf/UML.pdf>) så vi hvordan UML kan brukes i utviklingen og til dokumentasjon av systemer. De grafiske framstillingene hjelper oss å forstå et system. Nå skal vi bruke Natural Docs til å lage HTML dokumentasjon av kildekode. Resultatet for eksempelet Terningspillet Syv som ble brukt i UML anvendt i IT-2 ser slik ut:



Du kan prøve det ut på <http://tip.no/skole/res/output/files/Syv-as.html>

En av fordelene med objektorientering er at deler av systemet kan gjenbrukes i andre prosjekter og av flere personer. Grensesnittet til klassene (API - *Application Programming Interface*) må da være kjent. Med denne HTML dokumentasjonen oppnår vi blant annet det.

Utgangspunktet for HTML dokumentasjonen er kommentarene i kildekode. For å lage HTML dokumentasjonen er det bare å kjøre en .bat fil. Det lille du trenger å vite om Natural Docs kommentarer blir forklart senere i dokumentet.

Kompetansemålene under Planlegging og dokumentasjon i IT-2 omfatter blant annet:

- *velge og bruke relevante teknikker og verktøy for planlegging og utvikling av IT-løsninger*
- *forklare hensikten med teknisk dokumentasjon og lage slik dokumentasjon for IT-løsninger, med spesiell vekt på å dokumentere grensesnitt mellom ulike delsystemer*



# Natural Docs

## Installasjon

Gå til <http://naturaldocs.org/download.html> hvor de er en lenker til ActiveState's ActivePerl og siste versjon av Natural Docs. Perl programmet installerer seg selv (.msi), men Natural Docs må du pakke ut og legge i en mappe, for eksempel C:\Programfiler\NaturalDocs.

## Kjøre Natural Docs

I eksemplene som følger har jeg lagt kildekoden (Syv.as, Kopp.as, Terning.as og Utsyn.as i mappen c:\arbeidsfiler. Jeg har også lagd undermappene c:\arbeidsfiler\output hvor dokumentasjonen havner og c:\arbeidsfiler\project for konfigurasjonsfiler. Kildefilene til eksempelet, hvor kommentarene er noe endret fra UML anvendt i IT-2, kan du laste ned fra <http://tip.no/skole/res/NaturalDocsEksempel.zip>.

syv.bat filen som kjører programmet ligger i C:\Programfiler\NaturalDocs og ser slik ut:

```
NaturalDocs -i c:\arbeidsfiler -o HTML c:\arbeidsfiler\output -p c:\arbeidsfiler\project -cs UTF-8 -r
```

```
syv - Notisblokk
Fil Rediger Format Vis Hjelp
NaturalDocs -i c:\arbeidsfiler
-o HTML c:\arbeidsfiler\output
-p c:\arbeidsfiler\project -cs
UTF-8 -r
```

Alt må være på samme linje

-i er input mappe

-o er output mappe

-p er project mappe

-cs UTF-8 er charset som takler æ,ø og å

-r bygger alt på nytt

Når du kjører det fra Ledetekst ser det slik ut:

```
C:\WINDOWS\system32\cmd.exe
C:\Programfiler\NaturalDocs>syv
C:\Programfiler\NaturalDocs>NaturalDocs -i c:\arbeidsfiler -o HTML c:\arbeidsfiler\output -p c:\arbeidsfiler\project -cs UTF-8 -r
Finding files and detecting changes...
Parsing 4 files...
Building 4 files...
Building 5 indexes...
Updating menu...
Done.
C:\Programfiler\NaturalDocs>_
```

Du kan oppgi en tittel, en undertittel og stokke om rekkefølgen på menypunktene i fila Menu.txt som ligger project mappen. Jeg valgte "Syv" som *Title*, "Terningspill" som *SubTitle* og flyttet "File: Syv (Syv.as)" øverst i menyen.

## Natural Docs nøkkelord

Natural Docs bruker egne nøkkelord som du legger inn i kommentarene. Kommentarene kan ikke være på samme linje som koden. Det er ikke mye du trenger å lære, men hvis du vil se den fullstendige veiledningen så er den på <http://www.naturaldocs.org/documenting.html>. For å gi kommentar til en variabel skriver vi eksempelvis

```
// int: runde
// antall ganger spilleren har kastet terningene
private var runde:int = 0;
```

og får ut

**VARIABLES**

---

**runde**

---

```
private var runde: int
```

antall ganger spilleren har kastet terningene

Vi kan også bruke var, string ,boolean osv, men det er viktig at vi setter in kolon (:) og skriver variabel navnet helt likt (skiller mellom store og små bokstaver) slik det står under. Tilsvarende for funksjoner:

```
// Func: kast
// Hver gang kast funksjonen kjøres så settes verdien til
// et vilkårlig tall mellom 1 og 6
public function kast():void {
    verdi = int(Math.random()*6)+1;
} // slutt metode kast
```

Gir

**FUNCTIONS**

---

**kast**

---

```
public function kast():void
```

Hver gang kast funksjonen kjøres så settes verdien til et vilkårlig tall mellom 1 og 6

Natural Docs bryter teksten selv så vi må legge inn en blank linje dersom vi ønsker linjeskift. Egen heading, som for eksempel Forfatter, får man til med en blank linje, så en linje med Forfatter: og så navnet på neste linje. Punktliste begynner med – eller o, og vi kan godt bruke /\* kommentar \*/ som vist på neste side.

```

/*
 * Class: Terning
 * Hvert Terning objekt har en verdi som representerer hvor mange
 * øyne som vises på terningen.
 */
public class Terning {
    private var verdi:int;

```

gir

## Terning

Hvert Terning objekt har en verdi som representerer hvor mange øyne som vises på terningen.

Dersom du ønsker å bruke samme syntaks som i Javadoc, se

<http://java.sun.com/j2se/javadoc/writingdoccomments/index.html>

Så er det også fullt mulig:

```

/**
 * Returnerer verdien til en bestemt terning i koppen.
 * (Denne teksten er formatert med Javadoc syntax)
 * @param terningNr angir en bestemt terning i koppen
 * @return antall øyne som vises på terningen
 * @Version 15.12.2009
 * @author Ola Lie
 */
public function getVerdi(terningNr:int):int {
    return _kopp.getVerdi(terningNr);
} // slutt metode getVerdi

```

gir

### getVerdi

```
public function getVerdi(terningNr: int):int
```

Returnerer verdien til en bestemt terning i koppen. (Denne teksten er formatert med Javadoc syntax)

#### Parameters

`terningNr` angir en bestemt terning i koppen

#### Returns

antall øyne som vises på terningen

#### Version

15.12.2009

#### Author

Ola Lie