FORSLAG TIL RAMMEVERK, BIBLIOTEKER OG KONVENSJONER

March 16, 2017

Peter, Håvar, Markus og Elias
Universitetet i Bergen
Informatikk

Contents

BYGNINGSPROSESSEN

Bygningsprosessen definerer vi ved hjelp av Gradle. Færrest mulige steg fra å laste ned prosjektet til det er ferdig bygget skal oppnås. Hvis vi får dette til på en god måte blir det mye lettere å sette opp og jobbe med prosjektene og vanskeligere å ødelegge prosjektet med uhell slik at det ikke kan kompileres.

Bygningsprosessen Gradle utfører vil gå omtrent slik:

- Kjøre FindBugs
- Kompilere Java-filene
- Kjøre JUnit
- Sette sammen nødvendige filer til en kjørbar JAR-fil (eller andre format hvis spillene skal bli eksportert til andre platformer)

KONVENSJONER

For å redusere individuelle problemer som oppstår på grunn av forskjellige miljø har vi bestemt oss for å effektivisere prosessen av å laste ned prosjektet, importere det til en IDE og bygge og teste det. Dette har vi gjort ved å automatisere så mye av arbeidet som blir gjort mot prosjektet som mulig.

I tillegg har vi valgt noen standarder som må bli fulgt for at koden og filene i prosjektet blir lettere å lese og gjenkjenne.

Navn på filer i prosjektet

Navn på filer i prosjektet, spesielt ressursfiler (lyder, bilder, osv...), skal være på engelsk med små bokstaver og understrek i stedet for mellom rom. Selv om selve filnavnene er på engelsk kan innholdet være på norsk. Unntak av disse reglene kan forekomme hvis et verktøy forventer et spesifikt filnavn.

Noen eksempler:

- Et bilde av en basketball: *basketball.png*
- Et bilde av en trollfiende: *troll_enemy.png*
- En lydfil av et smell: *loud_bang.ogg*

Googles kodekonvensjoner for Java

Vi har valgt å bruke Google sine kodekonvensjoner for Java. Oracle sine var utdatert og manglet konvensjoner for nyere funksjoner i språket. I tillegg skal all kode og navn på kildekodefiler være på engelsk.

Googles kodekonvensjoner for Java

Format og lesing av filer

Formatet og lesing av eksterne filer burde også være standardisert slik at det er likt gjennom hele prosjektet. Dette gjør også at vi kan gjenbruke klasser som leser filformat i stedet for å lage nye løsninger for hver fil. Forslaget vårt er følgende:

Filtype	Format	Leser	Forklaring
Små-medium størrelse bilder med gjennomsiktighet	PNG	LibGDX bildeklasser.	
Store bilder uten gjennomsiktighet	JPG	LibGDX bildeklasser.	Gjelder bilder som ikke trenger å bruke tapsfri komprimering.
Lydfiler (korte og lange)	OGG	LibGDX lydklasser.	
Rene tekstfiler	TXT	Java sin IO-klasser.	
Hierarkisk tekstfiler	JSON	GSON-biblioteket.	Disse type filer kan brukes til å definere informasjon til elementer i spillene uten å rote til koden. For eksempel navnet og egenskaper på forskjellige karakterer i et spill.
XML (også en hierarkisk tekstfil)	XML	W3 sine XML-klasser (følger med JDK)	Dette formatet skal bare brukes hvis nødvendig (for eksempel til YR sin værdata).

RAMMEVERK OG BIBLIOTEK

Produksjon

LibGDX

LibGDX er et omfattende rammeverk basert på Lightweight Java Game Library (LWJGL) for å utvikle spill til web, mobil og desktop. Det er laget for Java og består av et hovedbibliotek og flere mindre tilleggsbibliotek. Det har eksistert lenge og er veldig mye brukt og har derfor mye dokumentasjon og andre ressurser tilgjengelig på nett.

Kjernen til LibGDX tilbyr klasser som på det lave nivået tar seg av vindubehandling, brukerinndata, lyd, grafikk, filsystem og nettverkslogikk. Alle disse modulene er bygget opp på en slik måte at omtrent alle platformspesifikke operasjoner er skjulte og derfor kan utviklere enkelt bygge til forskjellige platform fra en kodebase.

Hjemmeside · Wikipedia

Box2D

Box2D er et fysikkbibliotek for C++ som simulerer stive kropper i 2D. Det kan simulere realistisk, kontinuerlig kollisjon mellom mangekanter ved å påvirke kroppene med krefter, gravitasjon og friksjon.

Takket være arbeidet til LibGDX finnes det en Java-port som i bakgrunnen bruker de kompilerte C++ filene. På grunn av dette kjører det bedre enn hvis det hadde vært kjørt i JVM.

Hjemmeside · Wikipedia

SLF4J

Simple Logging Facade (SLF4J) er en abstraksjon av Java sitt innebygde loggingrammeverk (*java.util.logging*) som blant annet gir utviklere mer kontroll over hvilke alvorlighetsgrader som skal bli rapportert.

Hjemmeside · Wikipedia

GSON

GSON er et Java-bibliotek av Google som tar seg av serialisering og deserialisering av Java-objekter og ren JSON.

Hjemmeside · Wikipedia

Testing

JUnit

JUnit er et rammeverk for testing av komponenter (enheter) i et program. Det er en sentral del av XP (Extreme Programming). Testing består blant annet av å:

- Teste om objekter er like
- Teste om vilkår er sanne eller falske
- Teste om objekter er null
- Forvente at unntak oppstår

Hjemmeside · Wikipedia

Mockito

Mockito er et bibliotek for enhetstesting ved hjelp av "⊠⊠ mock objects"⊠⊠ i Java. Det brukes til å simulere et objekts oppførsel i kontekst av programmet og kan derfor simulerere komplekse objekter i et kontrollert miljø for å teste egenskaper som er tungvinte å teste ved kjøretidsanalyse.

Hjemmeside · Wikipedia

Sonarlint

En plugin i Eclipse og IntelliJ som kontrollerer kodestil i kildekoden fortløpende mens den skrives. Gir tilbakemelding om reglene for god kodestil ikke blir overholdt.

Hjemmeside

FindBugs

Plugin i Eclipse og IntelliJ som gjennomfører statisk kodeanalyse over valgt kodebase og kontrollerer dette opp mot fastsatte regler med hensikt å oppdage "code smell".

Hjemmeside · Wikipedia

Andre verktøy

IDE

Eclipse eller IntelliJ må brukes for å være sikker på at alle verktøyene som blir brukt fungerer hos alle. Ideen er at alle som produserer kode forholder seg til samme programmeringsmiljø og overholder lik kodestil og får hyppig tilbakemelding (short feedback loop) fra de forskjellige verktøyene.

IDE plugins

Til både Eclipse og IntelliJ finnes følgende plugins:

- Sonarlint (statisk kodeanalyse)
- Findbugs (statisk kodeanalyse). Kan også inkluderes som en del av bygningsprosessen.
- JUnit (enhetstesting). Kan også inkluderes som en del av bygningsprosessen.
- Gradle (definisjon av bygningsprosess). Inkludert i nyere versjoner av Eclipse.