

”Gisse GIS”
- webbasert geografisk informasjonssystem ved hjelp
av Open Source-programvare
TBA 4251 - Programmering i geomatikk, høsten 2012

Steffen Pøhner Henriksen

23. desember 2012

Sammendrag

Gisse GIS er resultatet av min innsats i faget TBA4251 - Programmering i geomatikk". Det er et enkelt geografisk informasjonssystem som kjører i en nettleser. Ved hjelp av Open Source-programvare vises vektordata over fritt tilgjengelige bakgrunnskart. Vektordataene kan så manipuleres med metoder kjent fra kraftfulle GIS som er desktop-applikasjoner. Metodene som støttes i dag er:

- Buffer - Utvider valgt område med en gitt verdi i meter.
- Area - Gir arealet av området i kvadratmeter.
- Merge - Slår sammen to vektorlag.
- Subtract - Trekker et vektorlag fra et annet.
- Intersect - Returnerer overlappende vektorlag.
- Distance - Finner minste avstand mellom to vektorlag.
- Simplify - Forenkler geometrien til et vektorlag ved Douglas-Peuker.

Programmet består av tre deler - database, serverlag og klient. Kommunikasjonen mellom server og klient skjer via Websockets[?]. Klienten er skrevet i Javascript, med HTML5 og CSS3. Tilleggsbiblioteker som er brukt i klienten er Knockout.js og jQuery, samt Twitter Bootstrap. Serverlaget er skrevet i node.js som tilbyr Javascript til bruk for serverkode. Her er sockets.io, Express og database-driver brukt som tilleggsbiblioteker. Eksempeldata brukt i utviklingen og til demonstrasjon er hentet fra Open Street Map og lagres i en PostgreSQL-database med PostGIS-utvidelse. Alt er lagret i skyen hos Amazons EC2-tjeneste, som sørger for at programmet leveres til brukers nettleser.

Resultatet er et brukervennlig GIS som utfører enkle operasjoner vi kjenner fra før, på frie vektordata som eksempel. "Gisse GIS" kan prøves ved å besøke adressen <http://gisse.pohnerhenriksen.com> i din nettleser.

Innhold

1 Innledning

Denne rapporten omhandler arbeidet med "Gisse GIS" utført høsten 2012. Den er et resultat av faget "TBA4251 - Programmering i Geomatikk" ved NTNU. Denne rapporten argumenterer for valg av programmeringsspråk, tilleggsbiblioteker og fremgangsmåte i arbeidsprosessen. Dataflyt og strukturen av programmet blir lagt frem. Et forslag til et scenario for å teste programmet blir også presentert. Oppdagede bugs, feil og mangler blir listet opp og kommentert.

Man kan dele inn arbeidet i fire deler - design, implementering, bugfiksing og rapportskriving. En stor del av arbeidet med utarbeidelsen av programmet var definere hva det skulle bli, hvilke verktøy som var hensiktsmessig å bruke, samt lære seg disse verktøyene. Tidlig ble det bestemt at jeg ville velge GIS-oppgaven i faget. Det gav meg muligheten til å jobbe med Javascript/HTML, noe jeg ikke har brukt i noen særlig grad før og har lyst til å lære meg. Oppgaven er farget av at jeg i størst mulig grad ville benytte meg av Open Source-programvare, og gjøre meg kjent med hva som finnes av slik programvare for bruk i geomatikkfaget.

Software is like sex: it's better when it's free." – *Linus Torvalds*

Tidligere har jeg jobbet hos Norkart AS som systemutvikler. Her laget jeg en database/server-løsning som behandlet geografiske data med tilleggsinformasjon. Kunnskapen fra denne sommerjobben kom godt med i utarbeidelsen av oppgaven. Kunnskapen opparbeidet herfra gjorde at jeg raskt så muligheter som PostGIS gir, og at implementeringen av databasedelen gikk relativt lett.



Figur 1: Arbeidsprosess

- 2 Formål med applikasjonen
- 3 Valg av programmeringsspråk
 - 3.1 HTML5 og CSS3
 - 3.1.1 Twitter Bootstrap
 - 3.1.2 Leaflet
 - 3.2 JavaScript
 - 3.2.1 jQuery
 - 3.2.2 Knockout.js
 - 3.2.3 Node.js
 - 3.2.4 socket.io
 - 3.3 PostgreSQL - PostGIS
- 4 Server i skyen - Amazon EC2
- 5 Testscenario
- 6 Bugs, feil og mangler
 - 6.1 Forbedringspotensiale
- 7 Diskusjon - Løser applikasjonen oppgaven?

Referanser

[1] Wikipedia: Websockets, <http://en.wikipedia.org/wiki/WebSocket>, 22.12.2012