

# Prosjektrapport

## Web-Prosjekt, høsten 2013

### Gruppe 7

Alf-André Walla, s181087  
Svein Gunnar Fagerheim, s181086  
Tore Angell Petersen, s181136  
Mateusz Zych, s190066

#### **Innledning**

Denne sluttrapporten omhandler det arbeidet som vi i gruppe 7 har lagt ned i løpet av prosjektperioden. Prosjektet er utført høsten 2013 ved Høyskolen i Oslo og Akershus. Gruppen består av fire personer, Alf-Andre Walla, Mateusz Zych, Svein Gunnar Fagerheim og Tore Angell Petersen. Der samtlige studerer Bachelor i Ingeniørfag, data.

I denne sluttrapporten vil det bli beskrevet: Hva som var våre forventninger. Hvordan vi tok tak i prosjektet. Hvordan vi jobbet med prosjektet. Hva som ble brukt for å oppnå ønsket resultat. Hva som ble implementert, samt med beskrivelse av sluttproduktet. Til slutt kommer det refleksjoner om prosjektgjennomføringen.

#### **Valg av tema**

Tema til vårt prosjekt var det ikke veldig mye spekulasjoner om. Det var ganske bestemt på forhånd hva er det vi skulle lage. En av gruppas medlemmer kom med prosjekt idéen før prosjektet kom i gang. Idéen var å lage noe lignende til «cookie clicker» spillet, bare med egen idé for tema. Alle likte ideen så vi bestemte oss å gå for den.

Da prosjektet kom i gang satt vi oss ned for å planlegge prosjektets fremgang. Der ble vi enige om at siden vi går data linje, så kan vi lage noe data relatert. Da kom idéen fra en av gruppas medlemmer om å lage det spille med tema knyttet mot programmering. Altså om å få flest mulig kodelinjer. Alle likte idéen og vi begynte å jobbe seg mot denne idéen.

#### **En detaljert beskrivelse av hva som er implementert, sett i sammenheng med forprosjektrapporten.**

#### **Use-cases**

Brukeren skal kunne trykke på en knapp = 1 kodelinje \* faktor, dette fører til at en kodelinje-teller økes for hvert klikk. Dette tallet øker over tid ettersom man oppgraderer og

kjøper ting som øker kodelinjer / sekund.

Ved å klikke på et keyboard som ligger oppå en Matrix bakgrunn kan spilleren øke antall linjer koder som har blitt skrevet. I begynnelsen av spillet vil det kun være mulig å få en linje med kode for hvert klikk. Etterhvert som man får kjøpt flere bygninger i spillet så vil det så vil antallet kodelinjer generert ved hvert klikk øke. Det er ingen grense på hvor mange kodelinjer som genereres for hvert klikk så det er fullt mulig å for eksempel generere en million linjer med koder for hver klikk, men det vil ta tid å komme så langt.

Skal kunne bruke antall kodelinjer til å øke automatisk produksjon av kodelinjer.

Kodelinjene man genererer kan man bruke til å kjøpe programmerere og andre ting som vil generere kodelinjer automatisk slik at spilleren ikke trenger å klikke på tastaturet hele tiden. Den koden som blir generert automatisk vil bli brukt til å kjøpe flere ting som vil øke antall linjer per sekund (cps) enda mer.

Skal kunne se statistikk over hva som er bygd, hvor mange kodelinjer brukeren har "skrevet," og mange andre ting.

Statistikken viser oversikt over dato for når spillet er startet. Hvor mange kodelinjer som er blitt skrevet totalt for alle spill, hvor mange kodelinjer som er blitt skrevet for dette spillet og hvor mange kodelinjer som er tilgjengelig for å kjøpe oppgraderinger med.

Statistikken viser også hvor mange kodelinjer man får hvert sekund av oppgraderingene man har kjøpt som kaller cps som står for codelines per second.

Det siste som statistikken viser er hvor mange museklikk man har brukt på keyboardet og hvor mange kodelinjer dette har generert.

Skal finnes en meny hvor brukeren kan resete spillet ved å trykke på en knapp

Det er blitt implementert to former for reset:

**Soft reset** som resetter det nåværende spillet. Dvs. alt av 'buildings', 'upgrades', 'codelines', 'clickpower', 'cps (codelines per secods)' og 'codelines (this game)' blir nullstilt. Slik at det som gjenstår er 'codelines (all time)'.

Mens **hard reset** resetter all data og statistikk.

Skal finnes en meny hvor brukeren kan lagre spillet ved å trykke på en knapp

Dette har ikke blitt implementert fordi det viste seg at denne knappen er unødvendig.

Siden nettstedet ble programmert slik at spillet blir lagret underveis hele tiden i realtime.

Skal kunne finne et easter-egg, forhåpentlig på en slik måte at hvem som helst skal kunne finne det hvis man leter.

Vi la til et morsomt easter-egg som passer til vår «nerdete» idé. Altså en gif av selveste Jim Carrey som pøser på tastaturet med 'codelines'. Så vis du spiller litt og har kommet deg til akkurat 1337 'cps'. Så dukker Jim Carrey gif'en opp og du får 500 000 'codelines' i bonus.

## **Tidslinje**

### Uke 44

Lage skjelettet til siden, og samtidig bestemme oss for et endelig design.  
Planlegging.

### Uke 45

Ferdig med planleggingsfasen.  
Implementasjon av spillet.  
Fokus på opplevelse og funksjonalitet.

### Uke 46

Begynne å skrive rapporten, fordele hvem som skal skrive hva.  
Skal være ferdig med det meste av spillet i løpet av denne uken.  
Polering av funksjonalitet og brukeropplevelse.

### Uke 47

Ferdig med spillet.  
Utarbeidelse av rapporten.

Siden vi tar opp dette faget så har det vært litt vanskelig å sammenstille jobbingen, siden vi går forskjellige år og vi som går samme år ikke har helt like valgfag så har vi hatt innleveringer i andre fag litt om en annen. Så da har vi ikke jobbet helt sammen, men med god kommunikasjon over skype så har vi klart dette ganske greit.

## **Evaluerings**

### Metoder

Vi begynte med en planleggingsperiode hvor vi fant ut hva som skulle være med. Deretter begynte vi å jobbe med hver vår del. Vi hadde ikke en formel gruppeleder som delegerte oppgaver. I stedet har det vært opp til den enkelte å finne ut hvilket arbeid som gjenstår, og ta tak i dette. Hvis det arbeidet man tok tak i ville ta lang tid så meldte man på Skype hva

man jobbet med slik at det ikke ble dobbeltarbeid. Hvis det var noen som slet med å finne arbeid så fikk dem hjelp til dette.

Ettersom alle fikk sin oppgave, så jobbet de med det til de ble ferdig. Alt mellom arbeidet ble lastet opp på Github. Slik alt alle har tilgang til enhver endring som ble gjort i prosjektet. Og hvis noe var uklart teknisk så har internett gitt oss mye på jakt etter informasjon og hjelp. Spesielt gjelder dette når det er ting som vi ikke har kunnet fra før, og som vi har hatt behov for å lære oss.

### Tester

Design:

Vi har utført noen brukertester på siden vår for å få relevant tilbakemelding om sidens opplevelse. Slik at vi kunne forbedre designet på siden. Dette viste seg å ikke være så lett, men til slutt etter mange forsøk så fikk vi det til. Mange av testere sa at siden så ikke helt fint ut, men hadde ikke noe særlig spesielt input på hva er det som kan forbedres. Ikke alle av disse personene var IT relaterte. Problemet gjaldt for det meste høyre delen av siden. Altså 'buildings' - knappene, men også generelt om siden skrift, kontrast, plassering osv. Rent designmessig. Etter flere tester kom vi oss frem til to IT relaterte personer som ga oss gode konstruktive tilbakemeldinger. Dermed så siden mer leselig og mye bedre ut.

Kontrast:

Ettersom designet ble ferdig programmert så har vi testet det også med "Colour Contrast Analyser". Der vi for det meste bestod wcag's trippel-A (AAA).

Funksjonalitet:

Sidens funksjonalitet ble testet av oss selv. Hver gang ny funksjonalitet ble lagdt til så har vi sjekket om helheten fungerer som den skal.

### Verktøy

Hver av oss har brukt det verktøyet de er mest vant til. Til sammen så har vi brukt Notepad++/gedit/NetBeans til å skrive kode. Og Git til å holde styr på versjon kontroll av koden. Repoet ligger privat på GitHub, som er tilgjengelig for alle gruppemedlemmer. Dette har fungert bra for oss, siden alt ble mellomlagret både lokalt og på GitHub (etter commits). Så ingenting av koden gikk tapt noen gang. Altså disse verktøyene har funket utmerket for oss i dette prosjektet.

### Teknologier

Fra starten av prosjektet hadde vi en klar idé for hva er det vi har lyst å lage. Derfor hadde vi også god oversikt over hvilke teknologier vi skulle bruke. Selvfølgelig måtte vi bruke HTML5 og CSS for det er det som er hoved tema i faget. Men for å oppnå det idéen vår innebar så måtte vi i tillegg ha med PHP, JavaScript og JQuery.

## HTML5:

Vi brukte den nyeste standarden av HTML for å lage strukturen til spillet. Der vi brukte div tagger for å holde orden på elementer.

## CSS:

Nettstedet benytter seg selvfølgelig av CSS, for å gi siden en passende til temaet design. Vår CSS struktur består av fire CSS-filer (Se lenger ned under 'Struktur'). Dette ble gjort slik for at det skal være mer oversiktlig og for å holde en god struktur på koden. Hver CSS-fil har ansvar for sin del av siden.

## PHP:

PHP har vi for det meste brukt for å sette sammen siden. Altså for å 'include' de forskjellige php filene sammen. Samt for å reportere eventuelle php feilene.

## JavaScript:

Alt funksjonalitet på siden er JavaScript basert. Dette har vi brukt for at alt spillet skal skje på klient siden og ikke på serveren.

## JQuery (siste versjon):

JQuery har vi brukt for å ekklere kunne ha fått tatt tak i forskjellige html elementene i JavaScript.

## Struktur

### CSS

- index.css
- left-section.css
- middle-section.css
- right-section.css

### PHP

- header.php
- index.php
- page-main.php
- keyboard.php
- stats.php
- menu.php
- upgrades.php

- achievements.php

## JavaScript

- buildings.js
- canvas.js
- csp.js
- hover.js
- init.js
- keyboard.js
- stats.js
- upgrades.js
- easteregg.js
- achievements.js

## Hva fungerte

Mye tid ble spart ved å bruke JQuery. Den øker livskvaliteten til enhver JS programmerer. ;) Fant ut at å bruke kompositt ikoner sparte mye tid etter implementasjonen var gjennomført, for man slapp å lage samme ikonet der det var 'selected' og 'disabled'. I stedet ble bare selve ramme-bildet laget i normal, selected, disabled versjoner, og dermed holdt det å lage 1 versjon av alle ikoner.

Prøvde å fokusere på klasser og medlemsfunksjoner i enkelte tilfeller i JS-delen. Det fungerte fint og fikk koden til å være effektiv og lesbar.

## Problemer

Der var noen problemer med å få knapper og slikt til å kunne highlightes med andre bilder via hover, når man i enkelte tilfeller også skulle disable knappene. Samtidig var noen knapper kompositter der bakgrunnen var selve ikonet, og der var en ramme i en div foran slik at grafikk lett kunne gjenbrukes.

Hadde problemer med å få en objekt-instans inn i eventer, men Google forklarte at alle lokale variabler som er definert i samme funksjon som definerer eventen vil være tilgjengelig når eventen avfyres. Dermed var det bare å skrive var self = this; også var man i mål.

## Forbedringer

Vi har ett par ting vi ikke har fått tid til, og noen ting vi har funnet som ikke helt fungerte.

Vårt easter egg har en liten bug i seg der om du finner easter egget og bare refresher siden så kommer easter egget opp vær gang, siden det er en bonus du får med det easter

egget så får du den hver gang du oppdatere siden.

### Tidsbruk

Design: Siden alle på gruppa er mer kode interessert enn design interessert, så førte det til at vi måtte bruke en god del til på dette. Design av siden var det vanskelig å bestemme. Til slutt ble det at vi gikk med programmering design siden prosjektet vårt handlet om dette. Da ble en av gruppas medlemmer inspirert av "matrix" -filmen han så i barndommen. Derfor ble det svart-grønn design, som passet veldig bra til temaet. Men fortsatt, selv om det var bestemt, så var det vanskelig å få det til å se bra ut. Spesielt når det gjaldt knappene på siden. Der gikk det mye tid.

### **Refleksjoner**

Møtte ingen utfordringer. Ingen timeplan eller estimerer sprakk.

Tidslinje evaluering:

Denne tidslinjen har vist seg å fungere veldig bra. Vi hold oss perfekt overens med planen!

Når vi ser tilbake på valgene vi har gjort i løpet av prosjektet så vi kan med sikkerhet konkludere at vi har foretatt de riktige valgene ovenfor tema, innhold og funksjoner på nettsiden. Vi føler vi har klart å leve opp til de forventningene vi satte oss selv da vi startet og føler vi har vært realistiske ved de målene vi har satt med tanke på hvilket faglig nivå vi har vært på etter hvert som tiden har gått. Vi er også tilfreds med løsningene vi har foretatt for å fylle de kravene som er pensum.

Ambisjonsnivået vårt lå ikke for lavt for kompleksiteten på nettstedet vårt er over pensums forventninger. Vi siktet relativt høyt og synes vi nådde målet godt ved hard og konsentrert arbeid.

### **Kilder**

Kilder til originale bilder som ble brukt til 'buildings' på siden:

img0 : [http://openclipart.org/image/50px/svg\\_to\\_png/172702/computer-worker.png](http://openclipart.org/image/50px/svg_to_png/172702/computer-worker.png)

img1 : <http://2shirt.org/wp/wp-content/uploads/2013/05/No-Comment-HTML.png>

img2 : [http://niezalezna.pl/uploads/foto/2011/12/czapka\\_student\\_magister.jpg](http://niezalezna.pl/uploads/foto/2011/12/czapka_student_magister.jpg)

img3 : [http://d24w6bsrhbeh9d.cloudfront.net/photo/6818802\\_fbthumbnail\\_v3.jpg](http://d24w6bsrhbeh9d.cloudfront.net/photo/6818802_fbthumbnail_v3.jpg)

img4 : <http://bybaxter.wpengine.com/wp-content/uploads/2013/01/optkeyboard.png>

img5 : <http://www.creepfilm.com/themes/creepn/images/Creep-logo.png>

img6 : [http://upload.wikimedia.org/wikipedia/en/1/12/The\\_creative\\_assembly-logo.png](http://upload.wikimedia.org/wikipedia/en/1/12/The_creative_assembly-logo.png)

img7 :

[http://openclipart.org/image/50px/svg\\_to\\_png/35269/preferences-desktop-keyboard-shortcuts.png](http://openclipart.org/image/50px/svg_to_png/35269/preferences-desktop-keyboard-shortcuts.png)

img8 : [https://www.novec.com/About\\_NOVEC/images/US-Government-Clip-Art.JPG](https://www.novec.com/About_NOVEC/images/US-Government-Clip-Art.JPG)

Border bilde som skiller de forskjellige delene på siden:

<http://visenri.users.sourceforge.net/images/matrix-01.gif>

Easter-egg gif: <http://media3.giphy.com/media/bAplZhiLAsNnG/giphy.gif>

En del av grafikken er “hjemmelaget,” etter behov. Matrix effekten krever ingen eksterne assets.