
Design 2: Github Commit Graph

Groep B5

Job Huisman	10119647
Tim Meijer	11153431
Sangeeta van Beemen	10340521
Stijn Robben	10559558
Demi Vollemans	10313680

Part 1 - Analysis

De relevante attributen van deze representaties zijn de commits, users, branches en de tijdlijn. Wat niet zichtbaar is in de representatie is de commit sizes, maar dit vinden wij ook niet relevant. De grootte van wat geüpload is op Github is misschien wel interessant maar niet bruikbaar in deze grafiek.

Commits: Bullets laten zien wanneer er een commit toegevoegd is. Wanneer je erop klikt

Users: In de eerste netwerk grafiek zijn de users te zien wanneer er met de cursor over de submits bewogen wordt. Dan komt de naam, foto en reactie van de gebruiker tevoorschijn. In de 2e netwerk grafiek worden alle verschillende gebruikers die het bestand bewerken in een aparte kolom weergegeven helemaal links in het document. Door te klikken op de verschillende gebruikers kan direct gezien worden wat de gebruiker heeft aangepast met betrekking tot het masterbestand.

Branches: De branches zijn zichtbaar, wat het gemakkelijk maakt om te zien waar wat is gedaan.

Timeline: De tijd-as van de netwerk grafiek is chronologisch en geeft de datum aan wanneer er een bestand is aangepast of gesubmit. Ook kan er aan de lengte van de pijlen (parallel aan de tijd as) gezien worden hoe lang iemand het bestand heeft kunnen bewerken voor dat het weer terug gemerged werd.

Commit sizes: Niet relevant/niet direct zichtbaar

De verschillende rollen die gebruikers van repositories op Github kunnen hebben zijn: owner en contributors/authors voor niet commerciële repositories en owner, billing manager en member voor commerciële Github repositories.

Rollen

Voor niet-commerciële repositories:

Owner

- Overzicht houden
- Inzicht in deelname contributors

Contributor

- Zien wat er gebeurt na de commits gedaan door de contributor
- Zien of het werk eerlijk verdeeld is

Voor commerciële repositories:

Owner

- Overzicht houden
- Inzicht in deelname contributors

Contributor

- Zien wat er gebeurt na de commits gedaan door de contributor
- Zien of de werklast eerlijk verdeeld is

Member

- Zien hoe lang een project duurt om een inschatting te maken over soortgelijke projecten

Verbeteringen;

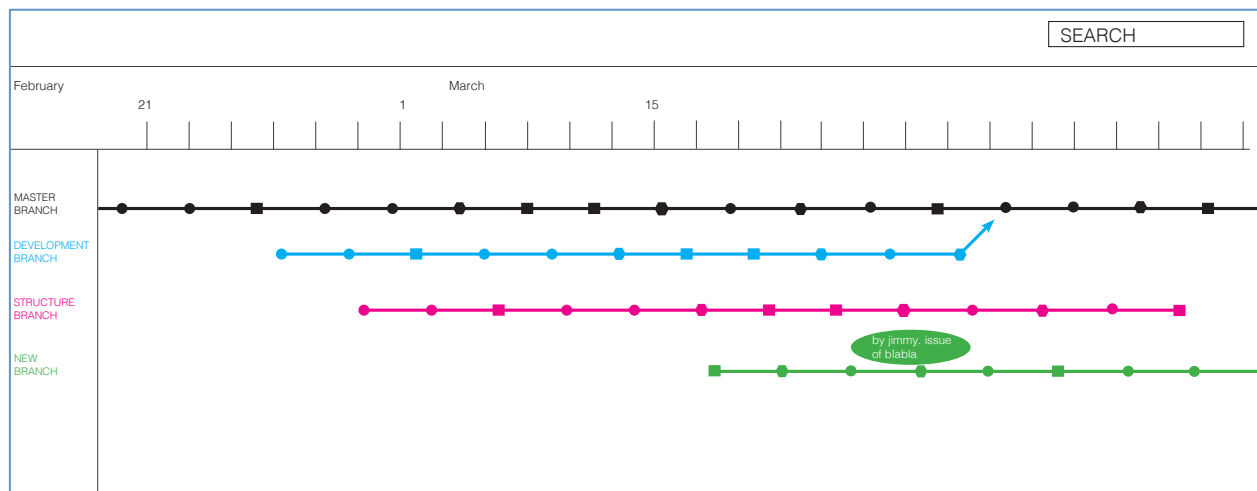
- Tijdlijn in-en uitzoomen (iedereen)
- Zoekfunctie naar commit, tijd (iedereen)
- Filters om enkel te weergeven wat elke contributor heeft gedaan (iedereen)
- Y-as verduidelijking, weergeven waarom omhoog, waarom naar beneden
- Weergeven wat voor soort commits er worden gepusht (iedereen)

Part 2 - Sketching

Schets 1

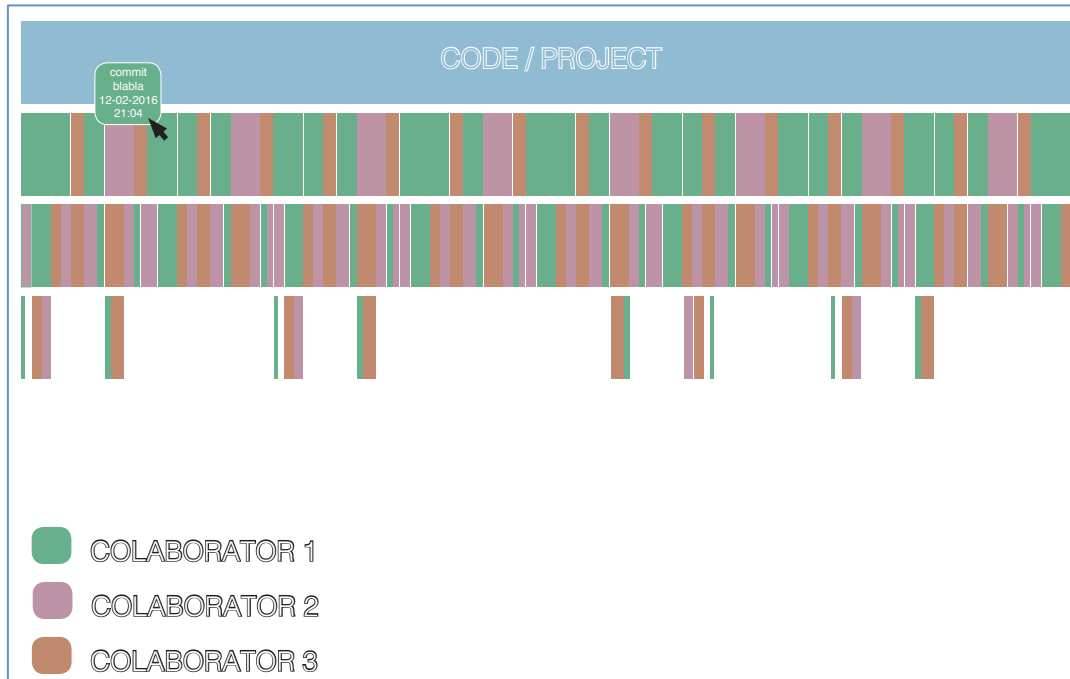
In schets 1 heeft men de optie om in en uit te zoomen op de tijdlijn.

Daarnaast is er de verbetering door dat er onderscheid in verschillende acties wordt gemaakt, deze hebben nu verschillende vormen, zoals een driehoek, vierkant etc. De type actie met bijbehorende vorm is terug te vinden in de legenda. Het figuur is ook interactief, als je met je cursor op een figuur gaat staan, verschijnt er een pop-up met meer informatie, over de desbetreffende actie, wie het heeft uitgevoerd, en exacte data. De y-as is verduidelijkt, op chronologische volgorde is aangegeven, van master naar member op data toegevoegd, wie welke aanpassing/opmerking heeft gedaan. Tot slot is er ook een search optie toegevoegd aan het figuur, als je een bepaalde actie of member zoekt, dan kun je dit gemakkelijk via de zoek-optie terug vinden, en is het niet meer nodig om door de hele tijdlijn te zoeken.



Schets 2

In deze schets is de bovenste balk de voortgang van het totale project. Onder deze grote balk ligt een laag die is opgedeeld in meerdere blokken, die laten zien hoe het project tot stand is gekomen. De kleur van het blokje toont welke contributor dit gedaan heeft. Als je met je cursor op het blokje gaat staan kun je zien wat voor contribution het is geweest (een aanpassing o.i.d.) en wanneer deze contribution is gedaan. De masterbranche zal bovenaan staan in deze visualisatie, met daaronder aparte visualisaties voor alle andere branches.



Part 3 - Group reflection

Objective

Github 2.0

De eerste sketch is vooral handig bij het snel krijgen van een overzicht m.b.t. informatie wanneer een bepaalde aanpassing gedaan is. Met behulp van de tijdlijn en de inzoom functie kan men zich snel oriënteren binnen de netwerkgrafiek en kan er snel en direct ingegaan worden op bepaalde specificaties van veranderingen. Daarbij zijn de branches op chronologische manier afgetakt, dus is de y-as veranderd en aangepast aan wanneer een bepaalde vertakking erbij is gekomen (**position**). Ook wordt er meer gebruik gemaakt van vormen als visuele variable. Door middel van de vormen (**shape**) wordt er aangegeven om wat voor een soort aanpassing het gaat (toevoeging, debug, deletie etc.).

Productivity scheme

De tweede sketch geeft beter weer hoe het script tot stand is gekomen door bijdrage van de verschillende contributors. Er is een duidelijkere weergave wat de hoeveelheid code van iedere contributor (**colour**) per programming step is (**size**). Dit is vooral handig bij het evalueren van de bijdrage en rol van de verschillende programmeurs in een project. Dus zowel voor school projecten als ook in het bedrijfsleven kan dit van waarde zijn.

Daarnaast is het ook mogelijk uit welke acties het script is opgebouwd door met de cursor over de verschillende programming steps te navigeren. Hiermee kan naast de kwantiteit ook de rol van een contributor onderzocht worden.

Na lang discussieren zijn we tot de conclusie gekomen dat sketch 1 voornamelijk gunstig is voor het grote publiek. Het snelle overzicht dat verkregen wordt kan goed en duidelijk geïnterpreteerd worden door de individuen. Deze sketch zal echter te kort komen wanneer men meer specifieke informatie nodig heeft over de productiviteit, hiervoor zal sketch 2 uitkomst bieden. Geen design is ideaal, het hangt ervan af wat de doeleinden zijn.