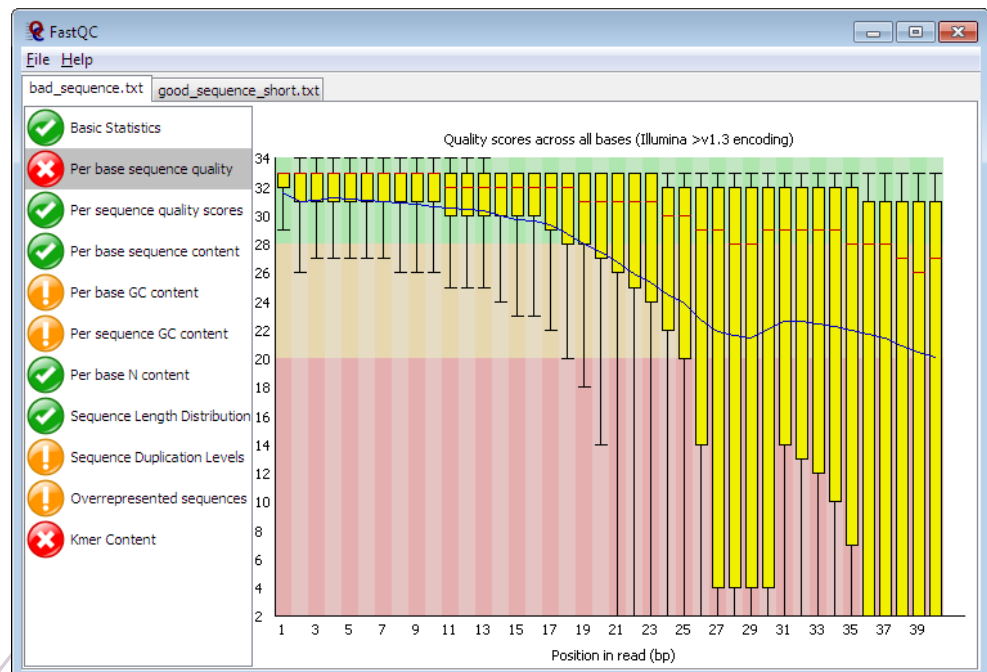


2/25/2025

Plan van aanpak

Practicum 3 / De BIN Toolbox



(Andrews, 2023)

FASTQC

Maartje van der Hulst

374361

CONTENTS

Inleiding	2
Achtergrond informatie	2
Project beschrijving.....	2
Project fases	2
Planning.....	5
Evaluatie en reflectie.....	5
Bronnen	6

INLEIDING

In dit project ga ik een HTML website maken waar een FastQC tool te gebruiken is door de gebruiker. Het gaat hierbij om een tool waar de gebruiker een fastq bestand kan uploaden. Hier worden dan verschillende controles over uitgevoerd. De uitkomsten hiervan zijn door de gebruiker te zien. Naast de tool die op de website te gebruiken is zal er ook pagina's met achtergrond informatie en dergelijke zijn. Ten slotte zal de gebruiker ook de optie hebben om de resultaten te downloaden en dus te bewaren.

In dit plan van aanpak zal ik de nodige achtergrond informatie uitwerken; een beschrijving van het project geven; de verschillende fases van het project beschrijven, wanneer die voltooid zijn en een grove tijdsplanning; en tot slot hoe de reflectie en evaluatie van dit project en product zullen gaan.

ACHTERGROND INFORMATIE

De FastQC tool voert kwaliteitscontroles uit over fastq-, SAM- of BAM-bestanden. Deze bestanden bevatten o.a. genetische informatie in de vorm van nucleotiden en een kwaliteitsscore per base, ook wel Phred Score waarbij een symbool (ASCII) aan een waarschijnlijkheid van onjuistheid verbonden is. (BAM File Format.; Sequence Alignment/Map Format Specification. 2009; Li et al., 2009) Deze bestandstypes komen voort uit Next-Generation sequencing. Dit is een proces van de sequentievulgorde vaststellen van een stuk genetisch materiaal, DNA of RNA (*DNA sequencing (article) | Biotechnology.*). Next-Generation sequencing is nieuwer, goedkoper, sneller en grootschaliger dan Sanger sequencing, wat voor het 'Human Genome Project' gebruikt is. (*Human Genome Project Timeline.*; Campen, Dr Amy Frost, Dr Julia van) De FastQC tool is vaak onderdeel van een pipeline en is dan de eerste stap na verkrijgen van de ruwe data. Het is een breed inzetbare tool, die voor veel verschillende soorten onderzoek nuttig is. Er worden door de tool 10 verschillende controles uitgevoerd, die de gebruiker voor bijvoorbeeld systematisch fouten of vervuiling van het sample kan waarschuwen. Aan de hand hiervan kan de gebruiker beslissen wel of niet verder te gaan met de data of dat deze verbeterd moet. De 10 verschillende controles hebben standaard parameters voor normale, licht abnormale en zeer ongebruikelijke resultaten. Maar voor verschillende projecten kunnen de wenselijke of logische parameters best veel uiteenlopen. Daarom is het belangrijk dat de gebruiker zelf de parameters kan aanpassen, voor een, voor dit specifieke project, accurate analyse en rapport. (Babraham bioinf, 2010)

PROJECT BESCHRIJVING

In dit project ga ik een website maken door middel van HTML, CSS, Flask, Jinja en Python. De website zal meerdere pagina's bevatten namelijk:

- Een homepage met een menu voor navigatie en een footer met relevante informatie.
- Een pagina waar de achtergrond informatie beschreven staat.
- Een pagina die het project toelicht.
- Een pagina waar de gebruiker haar bestand kan uploaden en de parameters voor de controles en bijpassende visualisaties kan instellen.
- Een pagina voor de resultaten doormiddel van de FastQC tool, die gedownload kunnen.

Deze website wordt interactief en krijgt nette error handling. Verder zal het project object georiënteerd geprogrammeerd worden (OOP).

PROJECT FASES

De opdracht is in 7 weekopdrachten uitgesplitst. Die houd ik aan als hoofdfases, maar die zal ik hier verder opdelen zodat het project in duidelijke stappen is opgedeeld.

Fase 1: Plan van aanpak. In deze week is het doel om een dataformat en een visualisatietool te kiezen. Die vervolgens verder uitzoeken en een plan van aanpak te schrijven voor dit hele project. Dit bestaat uit de volgende fases:

- 1.1 Verdiep je kort in de dataformats en kies er een uit met een bijpassende visualisatietool. Bespreek met je docent of dit een geschikte optie is.

- 1.2 Voer een achtergrondonderzoek uit naar het dataformat en de gekozen tool. Schrijf een kort stuk over de achtergrond, wie gebruiken de tool en waarom, wat voor onderzoeksvragen kun je beantwoorden met de tool ed. Zet een refworks op voor alle gebruikte bronnen.
- 1.3 Schrijf de inleiding en de projectomschrijving.
- 1.4 Zet het project uit in fases en maak de project planning, zorg hierbij voor duidelijke deadlines.
- 1.5 Ga na wat de momenten van evaluatie en reflectie moeten zijn. Zet uit wat de fase-doelen zijn en wat het projectdoel is. Beschrijf hoe het eindproduct geëvalueerd moet worden.
- 1.6 Zet het plan van aanpak netjes in elkaar. Zorg voor voorblad, inhoudsopgave en bronvermelding. Doe een spellingscontrole en lever het plan van aanpak in.
- 1.7 Download alvast de tool om hiermee bekend te kunnen raken. Zet een git, logboek en ticketing systeem op voor versie controle en takenoverzicht.

Fase 2: Statische website. Het maken van een statische website door gebruik te maken van HTML5 en CSS. De website moet uit meerdere pagina's bestaan met een navigatiemenu en een footer.

- 2.1 Schrijf een korte evaluatie over de vorige fase en verwerk eventueel gekregen feedback.
- 2.2 Maak een HTML5 home pagina met CSS3 styling in een extern stylingsheet document.
- 2.3 Maak een pagina die het project beschrijft. Maak gebruik van tekst uit het plan van aanpak en vul dit eventueel aan.
- 2.4 Maak een pagina die het gekozen format en de daarbij gekozen tool beschrijft. Maak gebruik van tekst uit het plan van aanpak en vul dit eventueel aan.
- 2.5 Voeg een footer met contact informatie toe aan de website.
- 2.6 Voeg een menu toe waarmee de gebruiker naar de verschillende pagina's kan navigeren.
- 2.7 Valideer online de gemaakte CSS en HTML bestanden. Repareer waar nodig. Controleer op voldoende comments en leesbaarheid.
- 2.8 Lever de CSS en HTML bestanden in d.m.v. een link van je online repo.

Fase 3: Dynamische website. Maak een pagina waar de gebruiker in een formulier instellingen voor de visualisatie kan invullen en naar de server kan uploaden.

- 3.1 Schrijf een korte evaluatie over de vorige fase en verwerk eventueel gekregen feedback.
- 3.2 Maak een readme.md voor het project waar de installatiehandleiding staat.
- 3.3 Maak een statische pagina voor het gebruikersformulier.
- 3.4 Ga voorbeelden Flask door en de W3schools tutorial voor HTML forms. (De workshops Flask Jinja is op 12 maart, wat na de deadline voor deze fase valt. Heb deze vorig jaar gevolgd, dus dit moet geen problemen veroorzaken.)
- 3.5 Maak doormiddel Flask en Jinja het formulier dynamisch.
- 3.6 Valideer online de gemaakte CSS en HTML bestanden. Repareer waar nodig. Controleer op voldoende comments en leesbaarheid.
- 3.7 Lever de Readme.md en je website in d.m.v. een link van je online repo.

Fase 4: Maak de resultaten pagina die, voor nu met dummy-data, de resultaten weergeeft. Gebruik matplotlib om gemakkelijk plaatsvervangende grafieken te maken.

- 4.1 Schrijf een korte evaluatie over de vorige fase en verwerk eventueel gekregen feedback.
- 4.2 Maak een statische resultaten pagina.
- 4.3 Maak met dummy-data en matplotlib de verschillende grafieken.
- 4.4 Maak nu dit wederom dynamisch zodat de gebruiker tussen de verschillende grafieken kan wisselen.
- 4.5 Maak de knop die de gebruiker de resultaten laat downloaden als zip.
- 4.6 Valideer online de gemaakte CSS en HTML bestanden. Repareer waar nodig. Controleer op voldoende comments en leesbaarheid.
- 4.7 Lever de resultaten pagina in d.m.v. een link van je online repo.

Fase 5: Maak de website backend. Gebruik OOP voor je datamodel en dataparser. Maak ook error pagina's dat de gebruiker vertelt dat en wat er verkeerd gaat en hoe dit evt. op te lossen is.

- 5.1 Schrijf een korte evaluatie over de vorige fase en verwerk eventueel gekregen feedback.
- 5.2 Maak een OO datamodel.
- 5.3 Maak een OO dataparser.
- 5.4 Maak error pagina's die de error en eventuele oplossingen beschrijven.

- 5.5 Maak de error handling en koppel dat aan de juiste error pagina's
- 5.6 Controleer of je python code aan PEP8 voldoet. Controleer op voldoende comments en leesbaarheid.
- 5.7 Lever de code in d.m.v. een link van je online repo.

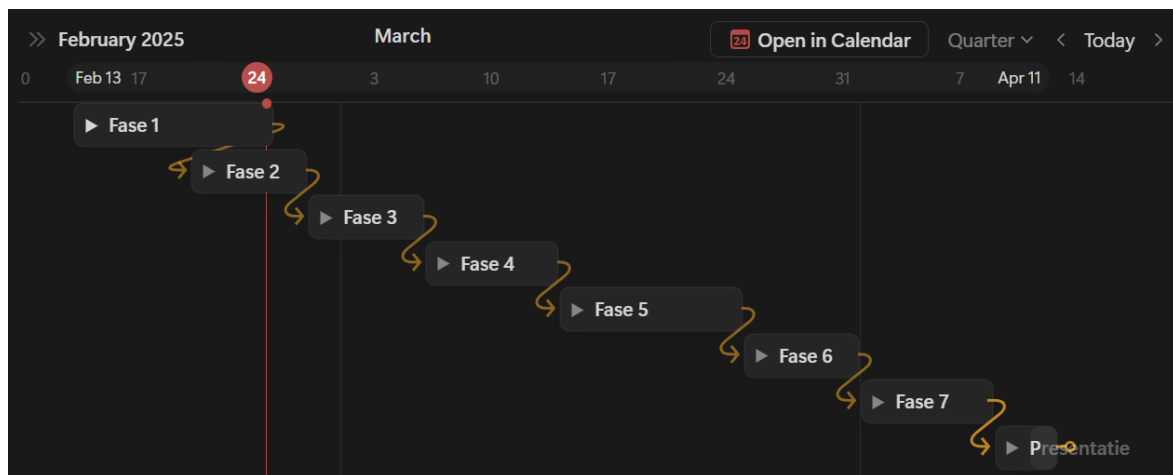
Fase 6: Koppel de backend en de resultatenpagina.

- 6.1 Schrijf een korte evaluatie over de vorige fase en verwerk eventueel gekregen feedback.
- 6.2 Koppel de backend en de resultatenpagina.
- 6.3 Zorg ervoor dat de formulierinstelling die de gebruiker invult gebruikt worden voor het maken van de resultaten.
- 6.4 Controleer of je python code aan PEP8 voldoet. Controleer op voldoende comments en leesbaarheid.
- 6.5 Lever de code in d.m.v. een link van je online repo.

Fase 7: Maak de website af en verwerk de gekregen feedback. Maak een eindpresentatie en presenteer die.

- 7.1 Schrijf een korte evaluatie over de vorige fase en verwerk eventueel gekregen feedback.
- 7.2 Neem alle gekregen feedback nogmaals door en pas alle relevante feedback toe.
- 7.3 Maak de styling van de website af.
- 7.4 Controleer of alle knoppen e.d. het goed doen en de resultaten te downloaden zijn.
- 7.5 Maak een lijst van alle delen die nog afgemaakt moeten of toegevoegd aan de website en werk die af. Ga hierbij de opdracht en beoordelingsrubriek nogmaals door.
- 7.6 Controleer of alle code aan de styling guidelines voldoen, pep8 voor python enz. Valideer de HTML en CSS documenten. Controleer op voldoende comments en leesbaarheid.
- 7.7 Neem het logboek, Readme.md, reflectie en evt. andere bestanden door voor spelling.
- 7.8 Schoon zo nodig je online repo op.
- 7.9 Lever alles in d.m.v. een link van je online repo.
- 7.10 Werk de vereisten lijst voor de presentatie door en maak de bijpassende presentatie.
- 7.11 Lever de presentatie in en doe je presentatie.

PLANNING



De deadlines die bij deze planning horen

Fase 1	25 februari 08:29
Fase 2	27 februari 08:29
Fase 3	6 maart 08:29
Fase 4	14 maart 08:29
Fase 5	25 maart 08:29
Fase 6	1 april 08:29
Fase 7	9 april 08:29
Presentatie	11 april 12:29

Mijn interne deadline voor de reflecties is dat die de dag van de deadline zelf geschreven moeten worden. Dat betekent dat de reflectie over fase 1 op 25 februari geschreven moet, enzovoorts. Dit geldt voor fase 1 t/m 6. Verder is er een peer feedback moment op 3 maart voor de installatie handleiding, dus dat is een deadline om de Readme.md af te hebben. Het peer feedback moment voor de dynamische website is op 19 maart en daarmee dus na de deadline van fase 3, dus hier is geen extra deadline voor nodig.

EVALUATIE EN REFLECTIE

Dit project heeft 7 weekopdrachten die beschrijven wat er in een fase afgemaakt moet. Hierdoor is er dus per fase een duidelijke doelstelling om naast het ingeleverde product te leggen voor evaluatie. Verder is er een beoordelingsrubric bij het 'de BIN toolbox'-vak, die voor een groot deel ook goed toe te passen is op dit project. Ook gelden er standaard regels voor alle ingeleverde code op de opleiding die vanzelfsprekend ook voor dit project van kracht zijn, zoals dat je code becommentarieerd is met docstring, block-comments en inline-comments of dat je code aan de PEP8 styleguide voldoet. In de project beschrijving en de project fases staan een groot deel van de specifieke eigenschappen van het eindproduct, namelijk de website, beschreven.

Voor het 'de BIN toolbox'-vak is er een specifieke reflectieopdracht over de samenwerking. Aangezien ik dit project alleen ga doen, zal ik aan het reflectieonderdeel een andere invulling geven. Na iedere deadline zal ik beginnen met een korte reflectie van max 250 woorden over de afgelopen fase. De vragen die bijvoorbeeld aan bod kunnen komen zijn: Heb ik de deadline gehaald?; Heb ik ontvangen feedback verwerkt?; Heb ik alle onderdelen van de opdracht gedaan?; Wat ging heel goed of viel je mee?; Wat ging slecht of viel je tegen?; Waar liep je tegenaan? Wat kun je verbeteren?; Heb je je logboek, git en ticketing netjes bijgehouden?. Deze reflecties zal ik per fase aan de repository toevoegen.

Andrews, S. (2023). *Babraham Bioinformatics - FastQC A Quality Control tool for High Throughput*

Sequence Data. <https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>

Using FastQC to check the quality of high throughput sequence. Babraham bioinf (Director). (2010, 9

22).[Video/DVD] <https://www.youtube.com/watch?v=bz93ReOv87Y>

BAM File Format.

[https://support.illumina.com/help/BS_App_RNASeq_Alignment_OLH_1000000006112/Content/Sou
rce/Informatics/BAM-Format.htm](https://support.illumina.com/help/BS_App_RNASeq_Alignment_OLH_1000000006112/Content/Sou
rce/Informatics/BAM-Format.htm)

Campen, Dr Amy Frost, Dr Julia van. *Sanger sequencing — Knowledge Hub*.

<https://www.genomicseducation.hee.nhs.uk/genotes/knowledge-hub/sanger-sequencing/>

DNA sequencing (article) | Biotechnology. [https://www.khanacademy.org/science/ap-biology/gene-
expression-and-regulation/biotechnology/a/dna-sequencing](https://www.khanacademy.org/science/ap-biology/gene-
expression-and-regulation/biotechnology/a/dna-sequencing)

Human Genome Project Timeline. <https://www.genome.gov/human-genome-project/timeline>

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., & Durbin, R.

(2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16), 2078–2079.

<https://doi.org/10.1093/bioinformatics/btp352>

Sequence Alignment/Map Format Specification. (2009). <https://samtools.github.io/hts-specs/SAMv1.pdf>