

Documentatie Roparun Tracker

Paul de Hek, 0941736
Rick van Vonderen, 0945444
Quentin Hoogwerf, 0929493
Wouter Bruggeman, 0943071

4 september 2018

Inhoudsopgave

| | | |
|----------|--------------------------------|----------|
| 1 | Inleiding | 2 |
| 2 | Code eisen | 3 |
| 2.1 | Classes | 3 |
| 2.2 | Methodes | 3 |
| 2.3 | Commentaar | 3 |
| 2.4 | Overig | 3 |
| 3 | Project eisen | 4 |
| 4 | Ontwerp | 5 |
| 5 | Reflectie | 6 |
| 5.1 | Paul de Hek | 6 |
| 5.1.1 | Wat was er leuk | 6 |
| 5.1.2 | Wat was er niet leuk | 6 |
| 5.1.3 | Wat was er moeilijk | 6 |
| 5.1.4 | Wat heb ik geleerd | 6 |
| 5.2 | Rick van Vonderen | 7 |
| 5.2.1 | Wat was er leuk | 7 |
| 5.2.2 | Wat was er niet leuk | 7 |
| 5.2.3 | Wat was er moeilijk | 7 |
| 5.2.4 | Wat heb ik geleerd | 7 |
| 5.3 | Quentin Hoogwerf | 8 |
| 5.3.1 | Wat was er leuk | 8 |
| 5.3.2 | Wat was er niet leuk | 8 |
| 5.3.3 | Wat was er moeilijk | 8 |
| 5.3.4 | Wat heb ik geleerd | 8 |
| 5.4 | Wouter Bruggeman | 9 |
| 5.4.1 | Wat was er leuk | 9 |
| 5.4.2 | Wat was er niet leuk | 9 |
| 5.4.3 | Wat was er moeilijk | 9 |
| 5.4.4 | Wat heb ik geleerd | 9 |

1 Inleiding

In dit verslag wordt beschreven hoe wij als team aan Project 0: Roparun hebben gewerkt. Het ontwerp, de projecteisen, code eisen en zelf-reflectie zullen allemaal aan bod komen.

2 Code eisen

In dit hoofdstuk staat beschreven wat de eisen aan de code zijn.

2.1 Classes

Het maken van classes is noodzakelijk omdat we Java gebruiken om de applicatie te ontwikkelen. Classes maken het ook mogelijk om code her te gebruiken.

2.2 Methodes

Een methode van een classe mag en kan maar 1 ding doen. Dit zorgt voor minder verwarring en duidelijkere methode namen.

2.3 Commentaar

Het gebruik van commentaar regels is een vereiste voor dit project. Een commentaar regel moet duidelijk beschrijven waarom de onderstaande methodes worden uitgevoerd.

2.4 Overig

Om te voorkomen dat er onvrede ontstaat door het gebruiken van snakecase en CamelCase, heb ik besloten om hier een advies over te geven.

Mijn advies is dan ook om gebruik te maken van CamelCase en alle methode en variable namen in het Engels te schrijven. Dit zorgt ervoor dat de code ook goed leesbaar is voor niet-nederlanders.

De code is te vinden op github: <https://github.com/wouterbruggeman/roparun-tracker>

3 Project eisen

- Ik als supporter wil zien waar het team op dit moment aan het rennen is, zodat ik als globale prestatie van het team kan bekijken. - Ik als supporter wil dat de applicatie mij een ETA (estimated time of arrival) bij een bepaalde locatie kan geven, zodat ik mijn spandoek met aanmoediging op tijd neer kan zetten. - Ik wil als runner later op maps kunnen inzien welke stukken van het parcours ik gelopen heb, zodat ik een persoonlijk overzicht en geschiedenis heb. - Ik wil als runner binnen elk gelopen stuk mijn data zien: snelheid, hartslag, weghelling, etc., zodat ik mijn sportieve prestaties per gelopen stuk kan analyseren. - Ik wil als runner mijn totale statistieken kunnen zien, om zo mijn sportieve prestaties te vergelijken of nieuwe doelen te stellen. - Ik wil als gebruiker dat de applicatie zo energiezuinig mogelijk is, omdat de accu 72 uur mee moet gaan, want tijdens de rustmomenten heb ik belangrijkere zaken om te doen, zoals eten, massagestoel bemachtigen en slapen. - Ik wil als gebruiker dat de applicatie veilig is, zodat ongeautoriseerde mensen niet bij mijn persoonlijke data kunnen. - Ik wil als loper dat de applicatie zelf herkent dat ik aan het lopen ben (ipv op een knop drukken om “activiteit” te starten), omdat ik stijf van de adrenaline sta en het in 8 van de 10 keren vergeet en daarmee worden de statistieken onjuist.

4 **Ontwerp**

5 Reflectie

5.1 Paul de Hek

5.1.1 Wat was er leuk

5.1.2 Wat was er niet leuk

5.1.3 Wat was er moeilijk

5.1.4 Wat heb ik geleerd

5.2 Rick van Vonderen

5.2.1 Wat was er leuk

5.2.2 Wat was er niet leuk

5.2.3 Wat was er moeilijk

5.2.4 Wat heb ik geleerd

5.3 Quentin Hoogwerf

5.3.1 Wat was er leuk

Het was leuk om voor een bestaand product iets voor te gaan programmeren. Het uitzoeken van de Wear OS API was ook interessant.

5.3.2 Wat was er niet leuk

Dat we te weinig tijd hebben om iets leuks neer te zetten, omdat veel tijd besteed is in het verdiepen van Wear OS

5.3.3 Wat was er moeilijk

Niet bekend zijn met het jargon van Android functionaliteit, wat het Google- en van oplossing vermoeijde.

5.3.4 Wat heb ik geleerd

- Git merge conflicts oplossen - Basic Android GUI - Permissies in Android via de manifest toevoegen

5.4 Wouter Bruggeman

5.4.1 Wat was er leuk

Het uitzoeken van de Android wear API vond ik erg leuk om te doen.

5.4.2 Wat was er niet leuk

Het is jammer dat het downloaden en en installeren van Android Studio erg lang duurde.

5.4.3 Wat was er moeilijk

5.4.4 Wat heb ik geleerd