

Plutus

Mobiele Toepassingen

UCLL 2015-2016

Kristof Dewilde

r0468690

Jan Pecquet

r0450688

3TX/4

Plutus

Meld je aan met je studentengegevens

LEES ONS PRIVACYBELEID

Studentennummer

r0468690

Wachtwoord

AANMELDEN

Wat is Plutus?

Plutus is gegroeid uit een reeks vragen waar geen enkele UCLL-student het antwoord op kent: heeft *die van de cafetaria* mij nu weer dubbel laten betalen? Als ik straks helemaal naar de automaat ga bij de aula, ga ik toch niet helemaal terug naar het oplaadpunt moeten omdat mijn saldo niet hoog genoeg is? Zijn er geen informaticastudenten die iets in elkaar kunnen steken daarvoor?

Vandaag heeft Plutus als een native app voor Android haar allereerste vorm aangenomen. Het ambitieuze idee is om op een dag liever vroeger dan later, een breed assortiment aan platformen te ondersteunen met een reeks apps die gebruiksgemak en snelheid op een proactieve manier centraal stellen, helemaal gebouwd rond het principe dat het geld op je studentenkaart jouw geld is en dat je toch wel mag weten wat er precies met je geld gebeurt.

Waar is Plutus nu?

Plutus verhuist binnenkort naar GitHub maar staat momenteel nog samen met de andere projecten van Kristof Dewilde op eenzelfde repository op BitBucket waarnaar ook Jan Pecquet gecommitt heeft. De andere taken van Jan Pecquet staan op een persoonlijke repository. U heeft leesrechten op beide vooraf genoemde repository's.

Voor de gemakkelijkerheid is er ook een APK van Plutus mee gecommitt geweest die gesigneerd werd via een keystore die toegang verleent tot de Google Maps API die in het project gebruikt wordt. Indien u het project zelf wil compileren, kan u de key voor de Google Maps API in de Manifest-file aanpassen naar uw persoonlijke key.

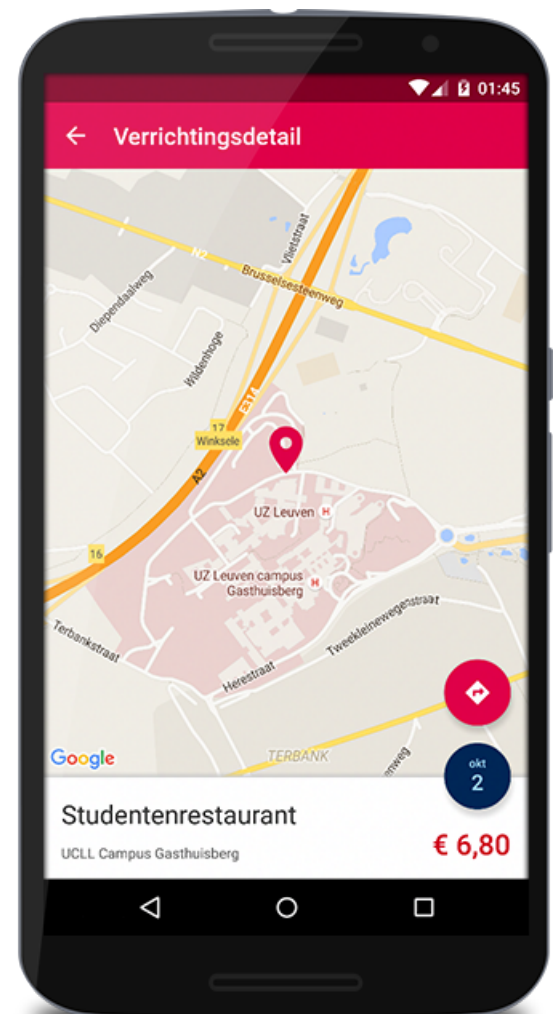
Zie het einde van deze bundel voor links naar onze Git-repository's.

Waarom Plutus?

Plutus is goed uitgewerkt. We hebben geprobeerd om de nuttige kennis uit vorige jaren te combineren met typische Android-programmeerpatronen. We droegen het *single responsibility*-patroon hoog in het vaandel en hebben heel wat gerefactored tijdens de uitwerking. We hebben ook de activity's en fragments gescheiden van de logica gebruik makend van *dependency injection* en abstractie. Het is zeker niet perfect maar we hebben een prima job geleverd.

Plutus is mooi dankzij een perfecte combinatie van de UCLL-huisstijl en Google's Material Design-specificaties. De app breekt op geen enkel scherm de opgelegde dimensies en zo goed als alle resources zijn afgezonderd -- @color, @dimen en @style. Hoewel Plutus' standaard-GUI gebouwd is met de resoluties van Android-smartphones vandaag indachtig, biedt deze consistente werkwijze een ideale omgeving om snel support uit te breiden naar nog meer toestellen.

Plutus is geanimeerd op subtiële plaatsen en voldoet hierdoor meteen ook aan een van de mogelijke extra functionaliteiten die in de app moesten zitten. Zo animeert de saldoteller van het vorige naar het huidige bedrag bij elke refresh, is het navigeren tussen activity's visueel logisch te volgen en dan spreken we nog niet over de animatie tussen de lijst met verrichtingen en de verrichtingsdetails, waar *shared element transitions* op een perfecte manier geïmplementeerd werd. Er is ook animatie voorzien voor



het verbergen van secties onder de instellingen die niet van toepassing zijn afhankelijk van de huidige voorkeuren van de gebruiker.

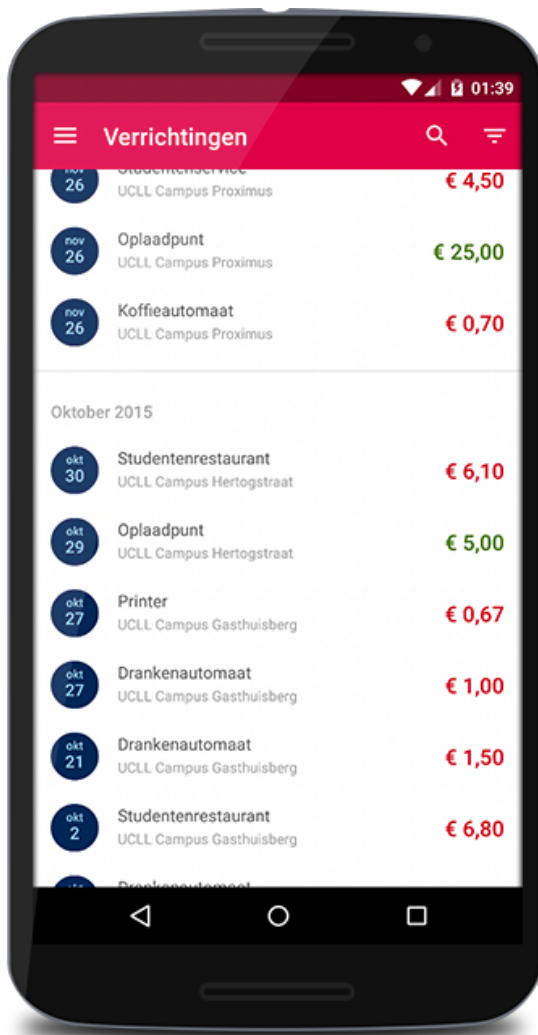


Plutus is snel, slim en asynchroon wat betreft dataverwerking. Van zodra het absolute minimum aan vereiste data gefetcht werd van de API, vernieuwt Plutus haar GUI indien de huidige, getoonde waarden verschillen van de nieuwe waarden. Plutus zal ook nooit onnodig data ophalen binnen een bepaalde logische termijn en zal automatisch de verbinding met de API sluiten wanneer geretournde data duplicaten zijn van reeds opgeslagen data.

Plutus maakt gebruik van een SQLite-database om verrichtingen lokaal bij te houden en voldoet zo nog eens aan een mogelijke extra vereiste. Plutus kan berekenen wanneer de API data teruggeeft die reeds in de lokale database aanwezig is en weet wanneer de database onvolledig is (bijvoorbeeld platte batterij tijdens fetchen). Plutus rekent ook op SharedPreferences voor simpele waarden die als key-value gestockeerd kunnen worden en zo heel snel uit te lezen zijn.

Plutus kan volledig functioneren zonder internetverbinding en schakelt zelf over naar een offline modus wanneer de app detecteert dat er voldoende lokale data aanwezig is voor een goede gebruikservaring. Hoewel een situatie zonder internetverbinding vandaag zeldzaam is (uitgezonderd niet functionerende Eduroam), is de offline beschikbare data de perfecte placeholder indien een nieuwe fetch onderbroken werd (bijvoorbeeld door het schakelen tussen 4G en 3G) of heel lang duurt (bijvoorbeeld op EDGE).

Plutus is, zoals het hoort, afgezonderd van de diepere logica en rust daarvoor op een eigen API die momenteel dummy data retournt, maar in de hopelijk nabije toekomst data kan halen van Intranet (begin dit jaar deed een gerucht de ronde dat informatie in verband met de verrichtingen van je studentenkaart “snel” beschikbaar zou komen op Intranet) of – in een ideale wereld – rechtstreeks van de UCLL zelf. Ook in dat laatste geval functioneert de eigen API als een vangnet voor wijzigingen dat data kan normaliseren naar een formaat dat de app verwacht. Dit voorkomt vele anders nodige kleine updates voor de app en staat toe om bepaalde logica die niet Android-specifiek is, buiten de Android-app te houden; klaar voor... een iOS-broertje? Android Wear? Windows Phone?



Plutus kan door al je transacties zoeken op een manier die bijna altijd sneller zal zijn dan om het even welke andere, doordat de app zo dicht bij die data zit dankzij de SQLite-database. Helaas zijn specifieke filters zoals definiëren van bepaalde campussen of een minimum- en maximumbedrag instellen nog niet geïmplementeerd in de huidige versie, maar zoeken op naam is wel al razendsnel mogelijk en bewijst dat een toekomstige implementatie geen bergen verzetten zal zijn.

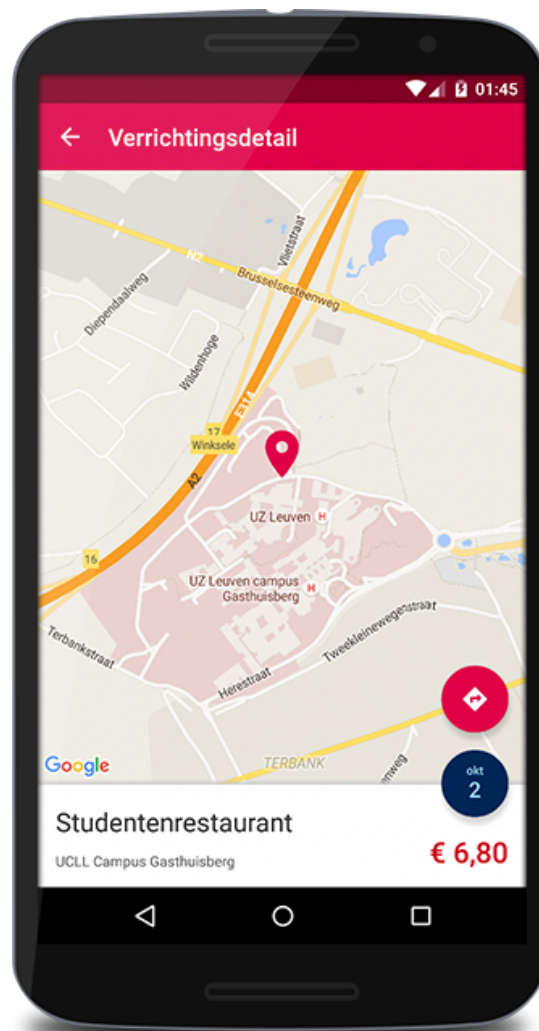
Plutus maakt gebruik van de Google Maps API om *super rich content* weer te kunnen geven bij de verrichtingsdetails en het is met één druk op een knop mogelijk om je favoriete navigatie-app te starten en te vertrekken naar de campus uit je verrichtingsdetails. Ook de kaart wordt uiteraard asynchroon gehaald en Plutus blijft functioneren wanneer op de telefoon van de

eindgebruiker Google Play Services of andere vereiste libraries ontbreken.

Plutus kan simpele HTML verwerken (zonder trage WebView) om verrichtingsdetails weer te geven. Dit maakt de connectie met de eigenlijke data veel lossier en zal voorkomen dat Plutus ongewenst reageert wanneer er anders onbekende data verwerkt moet worden.

Plutus is op het ergste voorbereid en staat de gebruiker snel en simpel toe om de database of zelfs heel de app te resetten voor in-geval-dat. Ook de nodige automatische verwijdering van data is voorzien zodat het mogelijk is om als een andere gebruiker te kunnen inloggen, zonder eerst heel de app daarvoor opnieuw te moeten installeren.

Plutus is persoonlijk dankzij persoonlijke voorkeuren die de gebruiker kan instellen. Toegegeven: vandaag is naast de saldoteller en het kiezen van het scherm waarmee de app moet starten, niet heel veel direct te personaliseren, maar de app biedt mogelijkheden.



Plutus ondersteunt verschillende talen en dat los van de op de telefoon ingestelde taal. In geval dat de gebruiker dat wenst, kan de app die anders wel de taal van de telefoon aanneemt, overschreven worden met een andere beschikbare vertaling: superhandig als je liever je telefoon in het Engels hebt staan maar UCLL-gerelateerde zaken in het Nederlands ziet, of je je Frans wil oefenen. De app is ook slim genoeg om datums te formatteren volgens de ingestelde taal (dd MMMM yyyy en MMMM, dd yyyy) en zelfs de euro's worden van de cents gescheiden met het symbool dat jouw taal van voorkeur daarvoor gebruikt!

En naast alle ingebouwde features, is Plutus vooral ook een applicatie met een reële context waar misschien niet direct heel veel vraag naar is, maar op elke telefoon zou staan vandaag, moest de dienst er al zijn. We kunnen alleen maar hopen dat de UCLL snel data beschikbaar stelt zodat we gemotiveerd blijven om aan Plutus te werken.

Git van Kristof Dewilde

`git clone https://Qrivi@bitbucket.org/Qrivi/mobileapplications.git`

Git van Jan Pecquet

`git clone https://Qrivi@bitbucket.org/JanPecquet/mobileapplications.git`

