



HOGESCHOOL ROTTERDAM / CMI

Mobile technologies & smart devices

CMTPRG03-7

Aantal studiepunten: 5
Modulebeheerder: Bas Boot



Modulebeschrijving

Modulenaam:	Mobile technologies & Smart devices														
Modulecode:	CMTPRG03-7														
Aantal studiepunten en studiebelastinguren:	<p>Dit studieonderdeel levert de student 5 studiepunten op, hetgeen overeenkomt met een studielast van 140 uren.</p> <p>De verdeling is als volgt:</p> <table> <tr> <td>Hoorcolleges</td><td>16 uur</td></tr> <tr> <td>Zelfstandige opdrachten + Vragenuur</td><td>30 uur</td></tr> <tr> <td>Online cursus</td><td>24 uur</td></tr> <tr> <td>Praktische opdracht</td><td>35 uur</td></tr> <tr> <td>Vorbereiding tentamen</td><td>30 uur</td></tr> <tr> <td>Deelname tentamen en feedback</td><td>5 uur</td></tr> <tr> <td>Totaal</td><td>140 uur</td></tr> </table>	Hoorcolleges	16 uur	Zelfstandige opdrachten + Vragenuur	30 uur	Online cursus	24 uur	Praktische opdracht	35 uur	Vorbereiding tentamen	30 uur	Deelname tentamen en feedback	5 uur	Totaal	140 uur
Hoorcolleges	16 uur														
Zelfstandige opdrachten + Vragenuur	30 uur														
Online cursus	24 uur														
Praktische opdracht	35 uur														
Vorbereiding tentamen	30 uur														
Deelname tentamen en feedback	5 uur														
Totaal	140 uur														
Vereiste voorkennis:	Object georiënteerd programmeren op niveau programmeren 4 (onderdeel van CMTTHE01-4) en ervaring met het werken met frameworks (CMTPRG02-5 en CMTPRG03-6)														
Werkvorm:	Online hoorcollege en zelfstandig werken aan opdrachten.														
Toetsing:	Praktische opdracht (beroepsproduct) en een online open boek tentamen (optioneel: online cursus)														
Leermiddelen:	Via Cumlaude Learning aangeboden lesmaterialen IntelliJ en Android Studio														
Draagt bij aan competentie:	<ul style="list-style-type: none"> • Technische kennis analyse (2) • Ontwikkelen & prototypen (2) • Testen en opleveren (2) 														
Leerdoelen:	<ul style="list-style-type: none"> • Ik begrijp geavanceerde object georiënteerde technieken, en kan deze toepassen: <ul style="list-style-type: none"> - Static / non-static class members - Encapsulation (Protection levels / Exception) - Polymorfisme (Inherritance, Interfaces, Abstract classes) - Collections • Ik ben in staat om zelfstandig informatie op developer sites te vinden, en kan de volgende mobile technieken uitleggen en gebruiken in een voor de gebruiker begrijpelijke en bruikbare toepassing: <ul style="list-style-type: none"> - Standaard user interactie, vormgeving en media, - User data bewaren op het device, - Informatie van internet halen, - De GPS locatie opvragen en tonen. 														
Inhoud:	In deze module leer je meer over object oriëntatie en technieken die je (o.a.) nodig hebt voor het ontwikkelen van native mobile applicaties.														
Opmerkingen:	Tijdens de lessen en opdrachten wordt er gewerkt met Android/Android studio. Studenten die voor een ander native platform (bijv iOS/XCode) willen werken kunnen dit doen in overleg met de docent.														
Modulebeheerder:	Bas Boot														
Datum:	April 2021														



1. Algemene omschrijving

1.1 Inleiding

Mobiele telefoons zijn al lang niet meer de traditionele apparaten die we alleen maar gebruiken om te bellen. Met de huidige generatie smart phones wordt bellen juist het minst gedaan en wordt in plaats daarvan het apparaat met de komst van allerlei handige apps voor de meest uiteenlopende doeleinden gebruikt. Door de moderne technologie in telefoons kunnen allerlei handige toepassingen ontwikkeld worden die ons leven makkelijker (of aangenamer) maken. Om deze apps te maken zijn ontwikkelaars nodig die kansen zien in de aanwezige technologie, de beperkingen van het betreffende platform kennen en met de platform specifieke SDK's kunnen werken.

In deze module komt de ontwikkeling van software voor een smartphone aan bod. In de les komen algemene principes aan bod, en ga je deze toepassen in Android.

1.2 Relatie met andere onderwijseenheden

Tijdens deze module wordt er vanuit gegaan dat de student alle programmeervakken uit het eerste jaar succesvol heeft afgerond, en minimaal één van de vakken CMTPRG02-5 of CMTPRG03-6.

1.3 Leermiddelen

Alle lesmaterialen (presentaties en lesbrieven, zowel directe informatie als links naar informatie) zijn te vinden via Cumlaude Learning.

Websites:

<http://developer.android.com>

Online cursus:

Udacity: Developing Android Apps (gratis)

2. Programma

Week	Theorie	Praktijk
1	Java basics, Strong typing, Encapsulation, overerving (herhaling)	Zie lesbrief
2	Interfaces, Abstract classes, polymorfisme	Zie lesbrief
3	MV VC principes UI / Layouts Mobile overwegingen Life cycle, hook-methods	Zie lesbrief
4	Geen les	Geen les
5	Rechten en privacy (werking)	Zie lesbrief



	User data en settings Enum Constanten Meertaligheid	
6	Internet / Json Exceptions Collections, Iterators	Zie lesbrief
7	GPS - wat en hoe Rechten en privacy (vragen)	Zie lesbrief
8	iOS vs Android / Play vs App Store Oefententamen	Zie lesbrief
9	Tentamen	Inleveren praktijkopdracht
10	Feedback	



3. Toetsing en beoordeling

3.1 procedure

Omdat deze module een theoretische component en een praktische component heeft zijn er verschillende deoltoetsen.

Opdracht

Je maakt een native mobile app waarin de technieken uit de module gebruikt worden.

De opdracht wordt uitgelegd tijdens de lessen, en wordt volledig beschreven op Cumlaude Learning.

Schriftelijk

Er is een online tentamen van 60 minuten over de theorie van deze module. Dit tentamen is 'open boek'.

Beoordeling

Het eindcijfer voor deze module is het gemiddelde van de opdracht en het schriftelijke tentamen. Om de module te behalen dienen beide cijfers 4,5 of hoger te zijn en het gemiddelde 5,5 of hoger.

De praktijkopdracht en het tentamen kunnen (apart) herkanst worden tijdens het volgende kwartaal.

3.2 data

De datum en tijd voor het tentamen worden gecommuniceerd via het rooster.

Inlevermomenten voor de praktijkopdracht zijn:

- 2 juli 2021 17:00 (eerste toetsmoment)
- 27 augustus 2021 17:00 (tweede toetsmoment / herkansing)

De methode van inleveren staat uitgelegd in de opdrachtomschrijving op Cumlaude.

Opdrachten die niet volledig of later worden ingeleverd worden niet beoordeeld.



Bijlage 1: Toetsmatrijs

	Leerdoelen	Taxonomie	Verwijzing naar opdracht/vraag/criteria
1.	Begrijpen en toepassen van geavanceerde OO technieken <ul style="list-style-type: none"> - (Static / non-static class members - Encapsulation (Protection levels / Exception) - Polymorfisme (Inherritance, Interfaces, Abstract classes) - Collections 	Fundamental Knowledge / Application	S
2.	Zelfstandig Informatie op de developer sites vinden	Learning how to learn	O
3.	Functionaliteit omzetten in voor de gebruiker begrijpelijke en bruikbare mobiele toepassing, zonder fouten of bugs	Application	O
4.	Een native mobiele applicatie kunnen bouwen (scheiding logica en views, comments in code)	Application	O
5.	User interactie, vormgeving en media gebruiken in een app	Fundamental Knowledge / Application	O, S
6.	User data bewaren op een mobile device,	Fundamental Knowledge / Application	O, S
7.	Informatie van internet halen	Fundamental Knowledge / Application	O, S
8.	GPS locatie gebruiken in een app	Fundamental Knowledge / Application	O, S
9.			
10.			

O = Opdracht

S = Schriftelijk tentamen