

Moduleteam: *Dirk Vandycke- Heidi Terryn - Thijs Martens - Kristien Roels - Corneel Theben*  
*Tervile - Jonas Maes – Jill VandenDriessche*

# Puzzle bobble 2015

## **Project Projecten II**

# 1 Situering van het project

Dit project kadert in de module Projecten II (3 ECTS punten)

Deze bestaat uit één enkele partim:

- Project (3 ECTS punten)

Zie ook de studiegids voor meer informatie

## 2 Beschrijving van de opdracht

Bubble Bobble anno 2015



Bubble Bobble is een klassieker van formaat die in de jaren '80 een ware rage was in de arcades op de Neo Geo spelconsole. Het is talloze keren geport geweest naar verschillende home consoles zoals SNES en PSOne, al dan niet onder dezelfde naam.

De hoofddoelstelling van deze opdracht is om aan de game “**Bubble Bobble**” een **opfrissing** te geven naar de noden van het jaar 2015. De basisprincipes van deze game blijven gelijk, maar moderne elementen zoals social media worden hierin mee verwerkt.

Er worden 4 zaken verwacht van jullie:

1. Een **analyse van de software** (die alle informatie en stadia bevat)
2. Een **werkende** (beperkte) versie van de **game** in **Java**
3. Een zelf opgesteld **ontwikkelingstraject** inclusief tussentijdse milestones
4. Een verzorgde **eindpresentatie** gericht naar mogelijke investeerders

## 2.1 Analyse van de software

### 2.1.1 Digitaal dossier - analyse

Er wordt een (digitaal) dossier aangelegd voor de ontwikkeling en analyse van deze game. De analyse bevat zowel de basisversie als uitbreidingen van de game. Enkel de basisversie dient geprogrammeerd te worden.

De gebruikte analysemethodiek(en) is/zijn vrij naar keuze, enkel **aan onderstaande minima moet worden voldaan**:

#### 2.1.1.1 Informatieanalyse

Het is de bedoeling dat een databank wordt opgesteld waarbij de informatie die in de casus beschreven wordt, vervat zit.

- 1) Opstellen van een conceptueel schema in ORM

Op basis van de beschreven opdracht stel je een conceptueel model op. Je gebruikt best een tool zoals Visio Modeler en maakt daar een project aan.

Probeer het conceptueel model indien nodig op te delen in logische delen die je per pagina kan weergeven i.p.v. het volledig model onoverzichtelijk op één pagina. Een object dat je reeds uitgebreid hebt beschreven, hoeft je niet telkens opnieuw te beschrijven maar duid je aan dat het reeds in een ander deel is beschreven (een zgn. extern object). Dit kan je in de tool aanduiden.

## 2) Mappen van het conceptueel schema naar het relationeel datamodel

Map het conceptueel model naar het relationeel model a.d.h.v. de Rmap-procedure. Het relationeel model geef je weer a.d.h.v. een ERD. Je mag geen tool gebruiken om het resultaat te bekomen maar uiteraard kan je deze wel gebruiken ter controle van het resultaat.

### 2.1.1.2 Databank

Het relationeel model met de nodige constraints en business rules wordt geïmplementeerd in MySQL. Je schrijft tevens de benodigde stored procedures, triggers en functies.

### 2.1.1.3 Wireframes

De volledige applicatie wordt uitgewerkt in low & high fidelity wireframes. Er mag geen enkele twijfel bestaan over wat er gebeurt “in geval van”. Dit is duidelijk in de wireframes.

(Tip: Indien je maar 3 schermen hebt in je wireframes ben je niet goed bezig...)

Deze documenten dienen als afspraak tussen jullie teamgenoten alsook de klant. Zie dit als onlosmakelijk onderdeel van het contract.

Je bent vrij van keuze voor een wireframing tool (potlood, pen, software), maar het moet vooral **LEESBAAR** zijn, ook de low-fidelity versie.

### 2.1.1.4 Functionele & niet-functionele Analyse

Je maakt een volledige functionele & niet functionele analyse van deze applicatie. Gebruik hiervoor het Requirements Analysis Document gezien tijdens de lessen van vorig jaar als leidraad.

We verwachten minimum:

1. Functional Requirements

1.1. Input

1.1. Output

1.1. Procesmanagement

1.1. Error Handling

2. Non-Functional Requirements

2.1. GUI requirements

2.1. Performance requirements

2.1. Security requirements

2.1. Other Non-functional requirements

Andere groeperingen zijn toegestaan, kies de vorm waarin je je het meest comfortabel voelt.

Use cases, slices en user stories zijn niet vereist, maar ze zullen je wellicht vooruit helpen bij het maken van de analyses.

Probeer van dit document vooral **iets nuttigs** te maken zowel voor jezelf als de klant.

Documentatie geschreven om de documentatie is voor niemand nuttig.

**Kwantiteit != kwaliteit**

## 2.1.2 Basisverwachtingen

### **1) *Gameplay concept single player:***

De speler beschikt over een “boog” die gecontroleerd wordt door de toetsen van het toetsenbord. Via deze boog kan men een bal afschieten in een bepaalde richting. De richting is beperkt tot  $160^{\circ}$  -  $20^{\circ}$

Bovenaan het scherm bevinden zich verschillende rijen “bubbles” in verschillende kleuren. De doelstelling is het wegwerken van alle bubbles door de kleur van de afgeschoten bal te matchen met de aanwezige bubble. Indien de kleur matcht, blijft de bubble “hangen”, tenzij 3 bubbles of meer van hetzelfde kleur zijn gematcht, dan verdwijnt de bijhorende cluster.

Als er onder een bepaalde geraakte cluster nog bubbels vasthangen, vallen die mee naar beneden. Zo kan je op een slimme manier grote blokken wegwerken.

Indien het scherm volledig gevuld is met bubbles en er dus een bubble de onderste rand raakt, is het spel voorbij.

Na verloop van tijd, door invloed van power ups / downs of een andere speler, kan het zijn dat het gehele speelveld opschuift naar boven of beneden. In dat geval komt een rij random bubbles binnen langs boven

De speler ziet onderaan telkens de volgende bubble die in zijn boog zal komen.

Verschillende bubbles geven verschillende punten op. Ook zitten er power ups of power downs verborgen in bepaalde bubbles (zie “power ups/ downs”)

De zijranden weerkaatsen de bubbels. Een bubble kan enkel via de onderzijde het scherm verlaten. De bovenzijde is standaard “sticky”, maar dit kan met moeilijkheidsgraad / power ups / ... beïnvloed worden.

We verwachten niet dat jullie principes fysica implementeren voor de animaties, maar wie zich geroepen voelt zullen we uiteraard niet weerhouden.

Een voorbeeld van de PlayStation Arcade versie:

<https://www.youtube.com/watch?v=79eTKHjJ19I>

## **2) Gameplay concept multi-player:**

De bubbles bevinden zich bovenaan het scherm. Net zoals in single player heeft speler 1 een boog aan de onderzijde van het scherm. Speler 2 heeft een duplicaat versie met zijn eigen speelveld ernaast (op hetzelfde scherm).

De acties van de ene speler beïnvloeden het speelveld van de andere. Als speler 1 bvb. een reeks “bubbles” wegwerkt, zakt het speelveld bij speler 2 met 1 rij (en er komen random nieuwe bubbles bij op die rij)

Indien bij 1 van beide spelers de bubbles de onderste lijn overschrijden, verliest deze speler het spel.

Bij multi-player spelen 2 spelers op éénzelfde toetsenbord op éénzelfde computer. Let erop dat je de controls ver genoeg uit elkaar plaatst.

## **3) Verschillende types bubbles:**

Niet alle bubbels zien er hetzelfde uit: ze verschillen minimum in kleur.

Er zijn **minimum 5 verschillende soorten bubbles die verschillen op kleur. Uiteraard mag je andere eigenschappen toevoegen.**

Let op: Deze gegevens worden bijgehouden in de databank.

## **4) Power ups:**

Voor het integreren van power ups en power downs is er artistieke vrijheid. Belangrijk is dat er **minimum 3 power ups en minimum 3 power downs** worden uitgewerkt.

Deze powers kunnen zowel om “bubbles” gaan als om afschietbare projectielen uit de boog.

Enkele mogelijkheden:

- Je kan de power ups verborgen laten voor de gebruiker en deze pas zichtbaar maken na het aanraken van een bubble of wegwerken van een cluster. Als 1 van beiden voldaan is, wordt de power up / down onmiddellijk toegepast (vergeet geen grafische feedback te geven)

Je kan op voorhand bepalen achter welke bubbles de power ups zich bevinden of dit random (eventueel gekoppeld aan de moeilijkheidsgraad) bepalen bij het inladen van het level.

- In plaats van de power up onmiddellijk toe te passen na het aanraken van de bubble / wegwerken van de cluster, kan je er ook voor kiezen om deze naar beneden te laten dwarrelen (dus wordt nog niet onmiddellijk toegepast). De gebruiker moet dan opnieuw een bubble afschieten en kiezen of hij deze gebruikt voor zijn speelveld of om de “power up” proberen te raken en zo te activeren

- Je kan ook power ups / downs op voorhand aan bepaalde kleuren van bubbles koppelen (bvb. alle aangeraakte blauwe bubbles activeren de “lazerbeam” powerup, alle zwarte bubbles zorgen ervoor dat er een nieuwe lijn bubbles bijkomt en het speelveld zakt) en deze met een legende per level aan de gebruiker tonen

- ...

Deze lijst bevat enkele suggesties en is niet limitatief!

Ook het type power up / down is aan jullie verbeelding overgelaten. Enkele voorbeelden kunnen zijn:

- Magic bubble: Bubble die alle bubbles van de kleur die hij aanraakt, doet wegvallen
- Rainbow bubble: Bubble die oorspronkelijk transparant is maar die een kleur overneemt van de eerste bubble die ertegen afgeschoten wordt
- Multiplier bubble: na het aanraken van deze bubble worden je volgende “dissolves” voor 30 seconden aan een multiplier score gerekend (x2, x4, ..)
- Bow Cannon: neemt een volledige rij ballen weg met een laserstraal
- ...

Brainstorm samen over mogelijkheden en doe inspiratie op op het web!

Let op: de parameters die je hier kiest, moeten kunnen worden aangepast in de database



### **5) Moeilijkheidsgraden:**

Er worden 3 moeilijkheidsgraden uitgewerkt: easy, medium en hard. Hoe je deze invult, kies je zelf.

Let op: de parameters die je hier kiest, moeten kunnen worden aangepast in de databank.

### **6) Score berekening en weergave:**

Het uiteindelijke doel van het spel is het uitspelen van de verschillende levels samen met het behalen van de hoogste high score. Telkens men een “level” vervolledigd heeft, start een nieuw level met meer bubbles, andere positionering en dus nieuwe scores te winnen. Je werkt 5 niveaus uit. Zorg ervoor dat je oplossing schaalbaar is naar 100 niveaus.

Aan het wegwerken van elke bubble is een score gekoppeld. Deze zijn gebonden aan een speler. In het geval van een 2 player game krijgen beide spelers individueel hun score. De score is te allen tijde zichtbaar op het scherm.

Bij de multi-player games zijn de scores individueel aan elke speler gekoppeld

### **7) Menu scherm:**

Een verwelkomingscherm geeft de gebruiker **minimum** de volgende mogelijkheden:

- Start game (moeilijkheidsgraad instellen!) + kies avatar
- Bekijk high scores single player (score + spelernaam)
- Bekijk high scores multi-player (wie haalt totaal aantal hoogste scores over al zijn multi player battles heen, dus tegen verschillende spelers)

## **2.1.3 Uitbreidingen**

Er is maar budget (en tijd) om de basisuitwerking van het spel in eerste fase uit te werken.

Echter na verloop van tijd zouden enkele uitbreidingen gewenst worden.

In het kader van gefaseerd ontwikkelen moet in het applicatiemodel reeds rekening gehouden worden met de volgende uitbreidingen:

### **1) Integratie social media**

Social media maken deel uit van de marketing campagne van een game. Denk na welke acties kunnen ondernomen worden om nieuwe spelers aan te trekken en bestaande spelers nog meer **verslaafd** te maken aan het spel.

**Een verplichte feature:** de speler moet na afloop van de game zijn **score** op social media kunnen **delen**.

Dit is echter niet voldoende voor goede social media integratie. Denk na over mogelijke interactie met andere mensen. Het moet meer zijn dan een gewone “share” button.

### **2) Clans**

Spelers moeten tot een bepaalde spelersgroep a.k.a. **clan** kunnen behoren. Die clans hebben een bepaalde ranking onderling. Denk na over of leden al dan niet beperkt moeten zijn en de voorwaarden om clans op te richten en te verwijderen. Ook het toetreden en verlaten van leden werk je uit.

Bedenk ook wat de motivatie zou kunnen zijn voor spelers om tot bepaalde clans te behoren (Exclusiviteit? Unlockables? In-game currency? Status? Gamification? Badges? ... )

### **3) Organisatie wedstrijden**

Op bepaalde tijdstippen moeten er wedstrijden tussen de spelers en clans kunnen georganiseerd worden.

Wie bepaalt de wedstrijden? Is er een beperkt aantal deelnemers? Kan dit over meerdere dagen doorgaan? Hoe wordt een ranking bepaald? Wanneer valt iemand af? Wat win je met een wedstrijd?

Dit zijn slechts enkele vragen, denk verder dan de bovenvermelde deelproblemen.

### **4) Campaign modus met tech tree**

Het spel moet naast de gewone single en multiplayer modus ook een campaign modus bevatten. Hierin worden een aantal voorgedefinieerde levels gevolgd die speciale regels kunnen toegepast hebben.

Tijdens deze campaign modus moet een “Tech tree” in voege zijn. Dat wil zeggen dat je na het verzamelen van een bepaald aantal “experience points (XP)” upgrades toegekend krijgt of kan kopen (vrij naar keuze). Deze upgrades neem je mee doorheen de campaign.

Je kan bvb. een level starten met een speciale bubble op je boog die je hebt aangekocht, de volgende 2 bubbles previewen i.p.v. 1, ...

### **5) Monetisation**

Het spel moet opbrengen voor de uitgever.

Werk uit hoe je hieraan geld gaat verdienen en verwerk dit in je ontwikkelingsmodel.

Advertenties kunnen deel uitmaken van je plan, maar zijn op zich niet voldoende.

Een overzicht van mogelijke inkomstenmodellen vinden jullie in een document op Leho.

## **2.2 Werkende (beperkte) versie van de game in Java**

### **1) Praktische uitwerking**

Je maakt een volledig werkende versie van de game volgens bovenvermelde basisvereisten in Java. Grafische elementen dien je niet zelf te maken maar mag je gebruiken van bestaande materialen. Denk wel aan bronvermelding!

Je werkt hierbij in teamverband volgens de aangeleerde principes van het object georiënteerd programmeren. Zorg dat je applicatie schaalbaar is waar gevraagd. Probeer aandacht te hebben voor het herkennen en toepassen van de principes achter design patterns en het refactoren in die richting.

### **2) Databank**

Je gebruikt een MySQL databank om de nodige gegevens in op te slaan. Kijk goed in de basisvereisten wat er minimum verwacht wordt bijgehouden te worden.

Gebruik je ER-diagramma uit de analyse ter ondersteuning, doch je werkt hier slechts een beperkt onderdeel van uit, namelijk **enkel het nodige voor de werking van de applicatie** (dus geen clans, wedstrijden, tech tree, etc.) beschreven in punt 2.1.2 “basisverwachtingen”

Voor het onderhouden van de database wordt **geen CMS** uitgewerkt, aanpassingen van de data gebeuren rechtstreeks in de databank zelf.

### **3) Unit testing**

De applicatie bevat de nodige unit tests. Het gaat hier vooral om de controle van het correct functioneren van de klassen. Een game simulator is niet nodig. Dit behelst ook integratietesten (om de goede samenwerking tussen klassen aan te tonen/te bewaken).

### **4) Versioning**

Voor versioning gebruik je git.

Maak een account aan op bitbucket en geef ook toegang aan [jonas.maes@howest.be](mailto:jonas.maes@howest.be) om je repository te bekijken. Let op: je kan slechts 5 gebruikers toevoegen in een gratis versie!

## **2.3 Ontwikkelingstraject**

### **2.3.1 Digitaal dossier – project management**

Deze opdracht bevat groepswork waarbij we de rollen in een bedrijf zo goed mogelijk zullen simuleren.

#### **Wat moet er zeker aanwezig zijn:**

##### **2.3.1.1 Toegewezen Takenlijst met deadlines**

Stel een takenlijst op (dit is uiteraard een dynamisch gegeven).

Gropeer je taken volgens fase:

Bvb. Analyse, ontwikkeling, testing, ...

**Wijs elke taak toe aan een teamlid.** Stel deadlines in per taak.

Indien mogelijk op het platform, stel een logische koppeling in tussen de taken.

#### 2.3.1.2 Milestones

Voeg een aantal belangrijke **milestones** toe. Controlemomenten waar je naartoe werkt.

Bvb. Analysebespreking, bètaversie klaar, presentatie, ...

#### 2.3.1.3 Kalender of GANTT-diagram

Zorg dat je een overzichtelijke weergave krijgt in de tijd van de uit te voeren taken, waarop ook de **milestones** zijn aangeduid. Dit kan in een calendar view of in een **GANTT-diagram**.

Dit diagramma dien je bij het indienen te exporteren via de software of via print screens indien de software dit niet toestaat.

#### 2.3.1.4 Planning vs eigenlijk gespendeerde tijd

Deel de grote fases analyse, ontwikkeling en testing op in voldoende onderdelen en **kwantificeer de geschatte tijd**.

Meet ook de effectief gespendeerde tijd per persoon en taak om na afloop van het project de juiste conclusies te kunnen trekken. Indien een bepaald onderdeel (dreigt) over tijd te gaan, wordt hier tijdig over gecommuniceerd.

**Alle afwijkingen van de oorspronkelijke planning en herverdelingen worden op gestructureerde wijze gedocumenteerd.**

We verwachten dat jullie werken met elkaar alsof deze situatie in een bedrijfscontext zou voorvallen.

Werk in je planning ook zogenaamde “**werksessies**” in. Dit zijn momenten waarbij je samenkomt om het analyseren / programmeren / ... door te nemen of uit te voeren.

#### 2.3.1.5 Kostenberekening & monetisation.

Beschouw 2 teamleden als ‘junior’ met een kostprijs €50 per uur, en beschouw 2 teamleden als ‘senior’ met een kostprijs van €80 per uur. Kies naast de kostprijs van de werkuren, zelf nog minstens 5 kostenposten die er volgens jullie in een reële situatie bij komen kijken en maak hier

een schatting van (huur, vervoerskosten, ...).

Bereken de geschatte kostprijs van het project op basis van de oorspronkelijke planning en vervolgens ook de uiteindelijke kostprijs van het project.

Herhaal hier ook de gekozen monetisation methode. Maak een inschatting van hoeveel omzet je zal genereren met het spel.

#### 2.3.1.6 Reflectie na afloop

Schrijf een reflectie waarin je zeker volgende elementen analyseert:

- Vergelijk de basislijn (oorspronkelijk plan) met het uiteindelijk uitgevoerde project. Is er veel gewijzigd aan de planning en toewijzing? Hoe groot is het verschil tussen de geschatte kostprijs en de uiteindelijke kost van het project?
- Is het plannen en de teamwerking goed verlopen? Hoe is de communicatie verlopen? Werden deadlines gehaald? Waarom niet?
- Is de kostprijs van het project te verantwoorden denken jullie of zou het efficiënter kunnen?
- Dekken de opbrengsten de kosten voldoende? Wat is jullie gerealiseerde winst/marge?
- Wat zouden jullie zeker behouden van jullie aanpak, wat zouden jullie in een volgend project anders doen?

#### **Tools:**

Maak indien gewenst gebruik van een applicatie zoals MS Project, of een cloud solution als Freckle, Trello of Basecamp om onderlinge communicatie te verbeteren.

Let op: Afhankelijk van het platform kunnen de (gratis) functionaliteiten verschillen of kan de licentie beperkt zijn in de tijd. In MS Project zullen jullie de meeste opties hebben.

## 2.4 Eindpresentatie voor mogelijke investeerders

### 2.4.1 Presentatie

Jullie krijgen bij het begin van de tweede projectweek een korte herhaling over presenteren en extra input over nonverbale communicatie.

Op basis hiervan maken jullie eerst een slidecast waarop jullie op het einde van de projectweek feedback krijgen door jouw medestudenten en door mevrouw Terryn.

Jullie houden rekening met deze feedback bij het opmaken van jullie definitieve presentatie. De definitieve presentaties vinden plaats op dinsdag 26 januari. De planning volgt via Leho.

Nog enkele praktische tips:

- Jullie presentatie duurt 10 minuten.
- Breng een samenhangend verhaal en probeer vooraf de vragen van het publiek in te schatten.
- Verzorg jullie voorkomen/jullie kledij.
- Attitude is uiteraard heel erg belangrijk. Een rinkelende GSM, kauwgum, ... het kan natuurlijk niet.
- Ook de manier waarop jullie de jury te woord staat is van belang! Begroet de juryleden, dank hen voor hun aandacht en geef rustig antwoord op vragen, ook al zijn ze misschien nogal direct geformuleerd.
- Probeer jullie werk te verdedigen als jullie voelen dat dit terecht is, maar wees ook niet te beroerd om eens toe te geven dat jullie een fout hebben gemaakt, of dat jullie niet meteen weten hoe je een vraag moet beantwoorden. Een zin als "Zo had ik het zelf nog niet bekeken, maar dat is zeker een waardevolle suggestie" is altijd handig voor in dit soort gevallen...
- Bezorg de jury hand-outs van jullie presentatie.

### 2.4.2 Evaluatieformulier presentatie

<b>A</b>	<b>Mijn tekst en inhoud zijn goed opgebouwd.</b>
	Mijn tekst heeft een driedelige structuur (opening, midden, slot)
	<b>Opening/inleiding</b>
	Ik trek de aandacht van mijn publiek, stel mezelf voor en begroet mijn publiek.
	Ik geef het onderwerp, thema en doel mee.
	Ik geef mijn structuur mee.
	<b>Midden</b>
	Ik houd mij aan de essentie.
	Ik geef functionele voorbeelden.
	<b>Slot</b>
	Ik koppel terug naar doel en doelgroep.
	Ik sluit krachtig af.
<b>B</b>	<b>Ik let op mijn (lichaams)taal en houd mijn stemgebruik in het oog.</b>
	Ik gebruik Standaardnederlands.
	Ik vermijd clichés en stopwoorden.
	De spelling op mijn dia's/hand-outs is correct.
	Ik articuleer en verzorg mijn intonatie.
	Mijn volume en tempo zijn aangepast.
	Ik ben enthousiast en dynamisch.
	Ik heb oogcontact met mijn publiek.
	Ik pas mijn gebaren aan.
	Mijn voorkomen is aangepast.
<b>C</b>	<b>Ik gebruik (audio)visuele ondersteuning.</b>



	Ik gebruik functionele hulpmiddelen.
	Ik gebruik een gepaste mix van audio-, beeld- en tekstmateriaal.

## 3 Doelstellingen van het project

Naast de competenties vermeld in de studiefiche zijn de doelstellingen van dit project verder driedig:

### **Doelstelling 1: Het kunnen uitvoeren van een informatiemodellering**

Dit bevat:

- Een informatieanalyse en -modellering kunnen uitvoeren, waarbij de bereikte resultaten geëvalueerd dienen te worden. Het bekomen resultaat kunnen implementeren in een relationele databank.

### **Doelstelling 2: Het kunnen programmeren volgens de principes van een gelaagd OO model in teamverband**

Dit bevat:

- In teamverband een gelaagde objectgeoriënteerde applicatie met een professionele grafische gebruikersinterface kunnen ontwikkelen. Deze applicatie maakt gebruik van een relationele databank om de gegevens op lange termijn te bewaren.
- De objectgeoriënteerde principes en de principes van gelaagd programmeren kunnen toepassen. Het herkennen en toepassen van de principes achter design patterns en het refactoreren in die richting.
- Het correct werken met een versiecontrolesysteem zoals git om het gemaakte werk onderling te kunnen onderhouden

### **Doelstelling 3: Het projectmatig aanpakken van een uitvoerbaar werk binnen een gegeven kader**

Dit bevat:

- Het opstellen van een eigen ontwikkelingstraject met eigen planning en tussentijdse milestones die de werkelijkheid weerspiegelen. De planning kadert in een raamwerk van opgelegde deadlines waaraan moet voldaan worden
- Het onderling toekennen van bepaalde rollen aan de verschillende teamleden en correct en tijdig uitvoeren van de toegewezen taken

## 4 De begeleiding van het project

### 4.1 Begeleiders

- Jonas Maes [Jonas.Maes@howest.be](mailto:Jonas.Maes@howest.be)
- Jill VandenDriessche [Jill.Vandendriessche@howest.be](mailto:Jill.Vandendriessche@howest.be)
- Dirk Vandycke [Dirk.Vandycke@howest.be](mailto:Dirk.Vandycke@howest.be)
- Kristien Roels [Kristien.Roels@howest.be](mailto:Kristien.Roels@howest.be)
- Corneel Theben Tervile [Corneel.Theben.Tervile@howest.be](mailto:Corneel.Theben.Tervile@howest.be)
- Thijs Martens [Thijs.Martens@howest.be](mailto:Thijs.Martens@howest.be)
- Heidi Terryn [Heidi.Terryn@howest.be](mailto:Heidi.Terryn@howest.be)

### 4.2 Organisatie begeleiding

Er zullen **coachingmomenten** worden voorzien waar een begeleider aanwezig is om advies te geven en het reeds gedane werk te evalueren en bij te sturen. Iedereen is steeds op die momenten aanwezig tenzij anders aangegeven. Voor alle administratie wordt de cursussite op Leho gebruikt.

## 5 Afspraken, coachingmomenten en deadlines

### 5.1 Coachingmomenten en deadlines

#### **Vrijdag 30 oktober**

*Wat:* Plenaire toelichting project en finale groepvorming (avond)

*Verplicht:* Ja

## **Projectweek 1**

*Wat:* Tussentijds feedback moment analyse / project management

*Verplicht:* Nee

### **Maandag 9 november**

*Wat:* Voorbereiding van de informatieanalyse.

*Verplicht:* Ja

*Wat tonen:* vanaf de middag zal een eerste check gebeuren van de bekomen resultaten met feedback

### **Donderdag 12 november**

*Wat:* Mondeling voorleggen planning en eerste versie analyse

*Verplicht:* Ja

*Wat tonen:* Digitale versie van het projectdossier met analytisch en project management luik.

Van het analytisch deel verwachten we een eerste versie (wireframes, basis FA, basis ORM).

Van het project management luik verwachten we dat jullie reeds een tool hebben gekozen en de toegewezen takenlijst (opgedeeld in fases) mét deadlines hebben aangemaakt. Ook de werksessies zijn ingepland en een aantal belangrijke milestones werden aangeduid.

### **Zondag 15 november**

*Wat:* Indienen eerste versie analyse

*Verplicht:* Ja

*Wat indienen:* Digitaal projectdossier met beide luiken vervolledigd na feedback. Gegeven feedback op 12 november is duidelijk geïmplementeerd.

### **Tussen 15 en 30 november (exacte deadline volgt)**

*Wat:* Indienen definitieve versie analyse na feedback begeleiders

*Verplicht:* Ja

*Wat indienen:* Digitaal projectdossier analyse volledig afgewerkt (dus ook mapping naar relationeel model), ontwikkelingstraject aangevuld waar nodig. Opnieuw is gegeven feedback duidelijk geïmplementeerd.

## **Zondag 13 december**

*Wat:* Indienen eerste versie werkend prototype:

*Verplicht:* Ja

*Wat indienen:*

- Java source files van je prototype Bij dit prototype gaat het om een stand van zaken, wij leggen geen tussentijdse verloopverwachting op. Hiervoor dient je eigen planning.
- Projectdossier met eventuele aanvulling / aanpassingen van de planning en/of projectverdeling

## **Projectweek 2**

*Wat:* Tussentijds feedback moment programmeren

*Verplicht:* Nee

## **Maandagnamiddag 14 december**

*Wat:* Input Presenteren en non-verbale communicatie

*Verplicht:* Ja

## **Donderdag 17 december 9.00 uur**

*Wat:* Droppen slidecast in Leho

*Verplicht:* ja

## **Vrijdag 18 december**

*Wat:* Indienen eindresultaat; peer assessment slidecast en feedback lector

*Verplicht:* Ja

*Wat indienen:*

- Java source files van je project
- Digitaal projectdossier volledig up-to-date
- Digitaal ontwikkelingstraject volledig up-to-date
- .jar executable van de applicatie
- Peer assessment (Leho)

## Dinsdag 26 januari

*Wat:* Eindpresentatie

*Verplicht:* Ja

Iedereen wordt verwacht op de presentatie. Jullie zullen elkaars werk zien en ook de feedback van de docenten horen.

**VOOR** de presentatie wordt door elk teamlid een peer assessment ingevuld waarbij hij zijn teamleden naar prestatie een score toekent. Deze scores worden niet gepubliceerd en zijn strikt vertrouwelijk.

Opgelet: Lokalen per sessie en eventuele wijzigingen in de planning worden via Leho gecommuniceerd.

## 5.2 Wijze van indienen

Gevraagde onderdelen zijn per deadline afhankelijk.  
Bij onderstaande is X telkens je groepsnummer

### 5.2.1 Hoe het projectdossier indienen?

Het projectdossier hanteert de volgende naamgeving:  
**projectdossier-groepX.pdf**

Voorzie een voorblad waar je de groep vermeldt en de studenten die deel uitmaken van de groep. Dit bevat enkel het **analytisch deel** van de opdracht.

### 5.2.2 Hoe de Java applicatie indienen?

De java applicatie zit in een map met de volgende naamgeving:  
**breakout-source-groepX**

Deze bevat de volledige mappenstructuur met bronfiles, libraries en mysql dump van de database.

### 5.2.3 Hoe de Java executable indienen?

De executable jar heeft de volgende naamgeving:  
**puzzlebobble-groepX.jar**

## 5.2.4 Hoe het ontwikkelingstraject indienen?

Het projectdossier hanteert de volgende naamgeving:  
**ontwikkelingstraject-groepX.pdf**

Voorzie een voorblad waar je de groep vermeldt en de studenten die deel uitmaken van de groep. Dit bevat enkel het **project management deel** van de opdracht.

# 6 Beschikbaar materiaal

- Lesmateriaal van de betrokken partims
- Leho cursussite
- Breakout documentatie: [https://en.wikipedia.org/wiki/Puzzle\\_Bobble](https://en.wikipedia.org/wiki/Puzzle_Bobble)
- Inspiratie powerups: <http://bbh.marpirc.net/bamagain/>
- Tools project management:
  - <https://basecamp.com>
  - <https://letsfreckle.com/>
  - <https://trello.com/>
  - <https://software.howest.be/software%20studenten/Microsoft>
  - <http://www.capterra.com/project-management-software/#infographic>
- Versioning: <https://bitbucket.org/> <https://git-scm.com/>

## 7 Evaluatie

### 7.1 Puntenverdeling

- productevaluatie: elk deel van de opdracht wordt gequoteerd (zie beschrijving van de opdracht onder punt 2). Het ontbreken of het onvoldoende uitwerken van een belangrijk onderdeel kan reeds een onvoldoende voor het project inhouden.
  - peer-assessment: het resultaat van het groepswork wordt vertaald naar een individueel resultaat volgens de individuele inbreng en medewerking.

### 7.2 Hoe gebeurt de evaluatie?

- Elk deel van de opdracht (analyse, project management en uiteindelijk eindproduct) wordt geëvalueerd volgens specifieke criteria. Tijdens de gemeenschappelijke feedbackmomenten

en na tussentijdse deadlines wordt door de begeleiders telkens feedback verzorgd waarvan implementatie noodzakelijk is.

De evaluaties resulteren in een groepscijfer voor het product.

- Op basis van de deelname aan de verplichte coachingmomenten, een peer-en co-assessment over de competenties, gekoppeld aan een functioneringsgesprek in groep kan het individueel cijfer afwijken van het groepscijfer, zowel in positieve als negatieve zin. Ongewettigde afwezigheid op één of meerdere coachingmomenten kan een uitzetting uit de groep impliceren en een nulquotering. Dezelfde sanctie kan genomen worden indien vanuit de groep voortdurend klachten komen dat één of meerdere studenten niet of onvoldoende meewerken. Uiteraard worden in dit geval alle partijen gehoord vooraleer een beslissing wordt genomen. **Hou er in elk geval rekening mee dat het project niet opnieuw gedaan kan worden in tweede zittijd (zie verder).**

### 7.3 Wat bij niet slagen in eerste zittijd

Gezien het project gecoacht wordt is een herkansing in tweede zittijd niet mogelijk. Er worden regelmatige feedbacksessies ingelast waarbij het resultaat tussentijds geëvalueerd wordt. De groep moet uiteraard met deze evaluaties rekening houden zodat de kwaliteit van het eindresultaat in positieve zin evolueert.