

Module :	PRE - Programming Expert		
Opdracht :	PE 2	Type :	Individuele Opdracht
Deadline :	Zie leho	Max score :	20

1 ALGEMEEN

1.1 DOELSTELLINGEN

- 2.1.3 Identificeert de verschillende actoren die betrokken zijn binnen het project
- 2.2.2 Weegt mogelijke infrastructuur af
- 2.2.4. Identificeert welke informatie ontbreekt en noodzakelijk is voor het project
- 2.2.5. Relateert de methodiek en doelstellingen aan de projectfasen en koppelt hieraan de actor die ze opstelt
- 3.2.9 Interpreteert een gemaakte analyse
- 6.2.2. Voorspelt mogelijke problemen in de specifieke softwareapplicatie

1.2 INDIENEN

- Dien jouw oplossing in via GitHub.
Documentatie plaats je in de **documentation** folder.

2 OPDRACHT 2 TEKST-BASED RPG

2.1 HET IDEE

Je wordt gevraagd om een text-based RPG game te maken. Het is momenteel nog onduidelijk hoe de verhaallijn er precies moet gaan uitzien, maar er zijn wel al een aantal ideeën die reeds uitgewerkt kunnen worden.

2.2 MINIMALE VEREISTEN

- Als speler wil ik een hero zijn of een hero kunnen kiezen wanneer een nieuw spel start;
- Als hero wil ik in een fictieve wereld ronddwalen en enemies aanvallen als ik ze tegen het lijf loop;
- Als hero wil ik in een fictieve wereld ronddwalen en weglopen van enemies als ik denk dat ik ze niet zal kunnen verslaan;
- Als hero wil ik bonus items en collectables kunnen verzamelen wanneer ik een enemy verslagen heb;
- Als hero wil ik bonus items kunnen activeren om mijn health, attack of defense skills te verbeteren;
- Als hero wil ik graag grotere bonussen krijgen wanneer ik moeilijkere enemies
- Als speler wil ik een end game scherm te zien krijgen

Denk er aan dat het om een text-based RPG gaat. Acties voer je als speler dus steeds uit door middel van vraag en antwoord. Je kan met andere woorden **niet** rondlopen. Bedenk zelf een kort scenario en werk een klassestructuur uit die het mogelijk maakt om dit na te spelen. Welke klasse (=gedrag/toestand) zou je hiervoor kunnen uitwerken?

Werk een klassendiagram uit waarbij je minimaal aan bovenstaande voorwaarden voldoet. Bedenk daarnaast nog 3 extra user stories die je uitwerkt.

Bonus items bestaan standaard uit bijvoorbeeld:

- extra health bonuses (bv: +5)
- extra attack bonuses
- extra defense bonuses

Maak bij de uitwerking van je schema gebruik van de nodige OOP structuren om het gevraagde game te kunnen modeleren. Je focus ligt initieel volledig op de core van je game: welk gedrag heb je en welke toestand hoor je daarbij te bewaren? Voeg pas later nice-to-haves toe aan je game.

Je bent vrij om naast deze basisfunctionaliteiten eigen extra's toe te voegen (=extra punten), maar zorg er eerst en vooral voor dat de basis aanwezig is. Extra's worden pas in rekening gebracht zodra de basisfunctionaliteit naar behoren werkt.

Hulp bij implementatie: Gebruik de UserInput klasse, die je kan terugvinden op leho, om informatie op te vragen van de gebruiker.

```
1. // eenvoudige tekst info opvragen UserInput userInput = new UserInput();
2. string name = userInput.ReadString("Well hello there stranger! What's your name?");
3. // Ja/Nee vragen stellen
4. UserInput userInput = new UserInput();
5. bool ynAnswer = userInput.ReadYesNo("Do you really think you're ready to fight this
   minion??");
6. // Getalwaarden opvragen
7. UserInput userInput = new UserInput();
8. int numberOfItems = userInput.ReadInt("How many items would you like to buy?");
9.
```