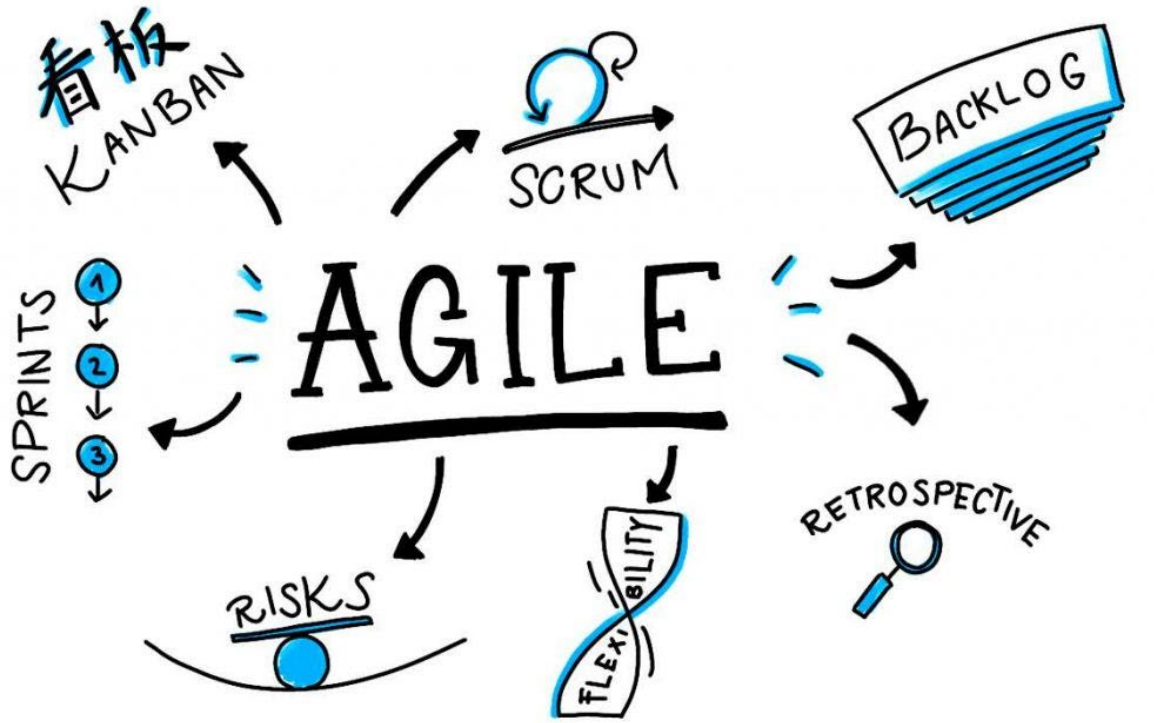
Project Agile Development



Deelproject:Cyber Security - Capture the Flag!
Datum: 30 december 2020, V1.0

Inhoud

1) Introductie

Tijdens Project Agile Development ontwikkel je een IT-product voor een echte opdrachtgever. Dit doe je met behulp van de Agile ontwikkelmethode Scrum. Scrum wordt in de beroepspraktijk veel gebruikt en is als zodanig onmisbare bagage voor elke IT'er.

Na afloop van Project Agile Development kun je:

- samenwerken met behulp van de ontwikkelmethode Scrum.
- een systeem modelleren, realiseren en testen binnen de ontwikkelmethode Scrum.
- een product bedenken dat aansluit op de eisen en wensen van de opdrachtgever/doelgroep.
- een product voorzien van een Engelstalige component.

Tijdens het project maak je gebruik van kennis en vaardigheden die je tijdens voorgaande studieonderdelen hebt opgedaan, daarnaast pas je de kennis en vaardigheden van de semester 2 vakken toe in dit project. Je werkt individueel en als team door middel van leerdoelen aan verschillende hbo-competenties.

Project Agile Development bestaat uit 5 deelprojecten. Deze projecthandleiding is bestemd voor teams van de leerroute Cyber Security. De handleiding beschrijft hoe het project is georganiseerd en wat er tijdens het project van jouw team wordt verwacht. Lees deze handleiding dus goed door en gebruik het document gedurende het project als naslagwerk.

2) Projectopdracht

Dit hoofdstuk beschrijft de opdrachtgever en de opdracht die je tijdens Project Agile Development uitvoert.

2.1 Opdrachtgever

De opdrachtgever van dit project is Leerroute Cyber Security HBO-ICT, vertegenwoordigd door de leerroute manager Patrick Boom, geadviseerd door CS docenten uit hogere jaars.
Gedurende de verdere ontwikkeling van de leerroute Cyber Security is meer en meer duidelijk geworden dat het noodzakelijk is dat er een omgeving nodig is waarin studenten en docenten Cyber Security threats kunnen ontwikkelen (training) en ontdekken. Een dergelijke omgeving traint de studenten in zowel het ontdekken, ontwikkelen en voorkomen van Cyber Security threats.
Voor een dergelijke omgeving is een Capture the Flag de meest aangewezen vorm. Door middel van gamificatie kan een studenten leren een threat te ontdekken en doorgronden en daarmee maatregelen bedenken om deze threats te voorkomen dan wel minimaliseren.
Tegelijkertijd is het mogelijk om een dergelijk CtF omgeving continue te voorzien van de laatst bekende threats en zo de studenten en docenten te trainen met de laatste inzichten.
Studenten en docenten kunnen (gecontroleerd) nieuwe challenges toevoegen aan deze omgeving.
Na enkele gesprekken is besloten om een dergelijke omgeving initieel te laten ontwikkelen door de studenten van CS jaar één in het project PAD.
De opdrachtgever zal gedurende het project nauw contact houden met de begeleiders van het project om ervoor te zorgen dat het project die producten oplevert die voor haar van belang zijn.
Tussentijds en aan het einde van het project zal een aantal van de studenten team producten aan de opdrachtgever gedemonstreerd worden en zal hierover feedback en een beoordeling geven.
Het is de bedoeling dat de producten aan het einde van het project door HBO-ICT leerroute Cyber Security in gebruik genomen worden.

2.2 Opdracht

Elk team ontwikkeld voor de opdrachtgever één of meer modules voor een Capture the Flag (CTF).
Elk team kiest één Cyber threat thema waarvoor zij de modules maken.

Een CTF is een wedstrijd waarbij een aantal deelnemers hun hacking skills kunnen laten zien in een gecontroleerde omgeving.

De organisatie (het development team) zet een omgeving op waarin zogenaamde flags gevonden kunnen worden, en de deelnemers moeten deze zien te verzamelen. Elke flag is punten waard, afhankelijk van de moeilijkheid, en aan het einde is het team met de meeste punten de winnaar.

Elk development team ontwikkeld zowel één front-end (website) als meerdere challenges waarin de challenges uitgevoerd worden en de flags veroverd kunnen worden. Vanuit het front-end worden de verschillende challenges gestart en worden de verkregen resultaten van de challenges verwerkt. Het is een wedstrijd dus goed registratie van de resultaten is noodzakelijke.

Het verzamelen van flags kan gedaan worden op verschillende gebieden, zoals het moeten exploiten van (web) applicaties (front-end of back-end code), het breken van zwak-geïmplementeerde cryptografie, steganografie, het analyseren van netwerkverkeer of netwerk forensisch onderzoek.

Omdat de CTF ook na het project gebruikt moet kunnen worden, is het belangrijk om goed te documenteren hoe de betreffende module beheerd en gebruikt kan worden, welke kwetsbaarheden aanwezig zijn en hoe deze kunnen worden gevonden.
Omdat elk development team onafhankelijk van de andere development teams werkt maar na afloop de verschillende challenges in één CtF omgeving geplaatst moeten kunnen worden is het belangrijk dat de afhankelijkheid tussen de front-end en de challenges minimaal is, duidelijk en volledig gedocumenteerd is en gemakkelijk te implementeren is in een ander front-end.

Van elke student wordt verwacht dat hij/zij zelf een volledige vertical slice kan en heeft ontwikkeld

Om zeker te weten dat de ontwikkelde CTF-module aansluit bij de doelgroep is het ook belangrijk deze door potentiële gebruikers te laten testen en een rapportage op te stellen over hun bevindingen. Dit kan door het organiseren van een red-team / blue-team met hulp van medestudenten van andere teams.

Wat is nu precies een CTF-module? Een aantal voorbeelden is (soms na registratie) te vinden op:

- https://ctf.hacker101.com/ctf
- https://ctflearn.com/dashboard
- https://overthewire.org/wargames/

3) Projectorganisatie

Dit hoofdstuk bevat de belangrijkste organisatorische informatie over het project.

3.1 Projectteam

Elk projectteam bestaan uit ± 5 tot maximaal 6 personen. Projectteams zijn zogenaamde zelfsturende teams, dat wil zeggen dat je gezamenlijk verantwoordelijk bent voor het totale proces waarin het projectresultaat tot stand komt. Het team bepaalt hoe de taken worden verdeeld. Als eis geldt wel dat er wordt gewerkt volgens Scrum en dat elk teamlid een aantoonbare bijdrage levert aan het maken van het eindproduct (zie ook beoordelingscriteria van zowel Product, Proces als Methoden & Technieken)

3.2 Projectdagen

(Online) Projectdagen
Voor het project zijn twee dagdelen ingeroosterd op maandagochtend (08.30-11.50) en woensdagmiddag (13.40 uur - 17:00 uur). Dit zijn gereserveerde uren om samen met je team te werken aan het project. Vanwege de corona maatregelen is dit online en niet in lokalen van de HvA
Op deze dagdelen werk je aan het project met je team.
Op Maandag werk je samen met je team aan het project (alleen studenten)
Op Woensdagmiddag is er ruimte voor workshops en begeleiding (Studenten en docenten).
Dit kan zowel tegelijk voor alle studenten van CS worden gegeven maar ook per klas/squad of team. De Tech- en/of Scrum/Team coaches kunnen ook van deze middag gebruik maken om speciale (online) bijeenkomsten en/of coaching voor het project te houden.

Oncampus project ondersteuning
Gedurende elke OnCampus bijeenkomst (mits toegestaan door de corona maatregelen en anders wordt dit online verzorgt) zijn er voor elke squad/team minimaal 2 lesuren gereserveerd.

1 lesuur gereserveerd voor Scrum/Team/Tech coaching
1 lesuur gereserveerd voor Team (begeleid) samenwerken

Aanwezigheid gedurende de projectdagen (zowel Online als OnCampus) is verplicht.

Naast de project dag wordt van elk teamlid verwacht dat hij/zij gemiddeld ongeveer 10 uur per week aan het project besteedt.

3.3 Begeleiding

Gedurende het project worden jullie geholpen door de volgende docenten:

Product owners
De product owner vertegenwoordigt in het project de opdrachtgever en is ook jullie vraagbaak op technisch gebied. De product owner:

- checkt elke sprint de backlog op inhoud en prioriteitstelling
- bekijkt elke sprint de opgeleverde (technische) documentatie
- biedt technische ondersteuning
- verzorgt naar behoefte korte workshops over technische onderwerpen

Agile coaches
De Agile coaches helpen de teams zich het werken volgens Scrum eigen te maken. De Agile coach:

- begeleidt teams bij het opstellen en 'pokeren' van de sprintplanning
- bekijkt elke sprint de backlog op goed gebruik van Scrum (bijvoorbeeld: goed geformuleerde user story's, realistische sprintplanning, in elke sprint wordt voor de opdrachtgever bruikbare functionaliteit opgeleverd.)
- doet elke sprint een keer mee met de stand-up
- begeleidt teams bij de retrospective

Team coaches
De teamcoaches ondersteunen de teams bij de samenwerking en coachen de individuele teamleden op het werken aan hun persoonlijke leerdoelen. De teamcoach:

- spreekt elke sprint met het team over de samenwerking binnen het team
- spreekt elke sprint met alle teamleden over de individuele leerdoelen en activiteiten om die te behalen
- ondersteunt teams bij problemen in de samenwerking

Elk team heeft de volgende docenten als eerste aanspreekpunt:

Team	**Product Owner & Tech Coach**	**Agile en Team Coach**
IC101	Gerald Stap	Theo Ris
IC102	Mary-Jo de Leeuw	Tiemen Mink
IC103	Krijn Hoogendorp	Kim Leiwakabessy
IC104	Maarten Bonsema	Esther Tielen
IC105	Karel van der Linden	Renske Siezen

3.4 Communicatie

Voor de generieke communicatie (communicatie voor alle klassen/squads/teams hetzelfde) zal gebruik gemakt worden van announcements in de DLO. Zorg er voor dat in je profile je notifications voor announcements aanstaan).
Zorg er ook voor dat je geregistreerd bent ik het juiste team in de groups van DLO.
Binnen een klas/squad of op team niveau kan een begeleider gebruik maken het versturen van email door middel van de DLO.
OnLine lessen en workshops die voor geheel CS tegelijk zijn zullen vanuit de Vritual classroomm van de DLO gegeven worden.
Begeleiders (tech of scrum) kunnen gebruik maken van virtual classroom en/of Teams volgens afspraken die de begeleiders met de klas/squad of teams maken.

Voor informele communicatie binnen teams en over teams heen (studenten) wordt gebruik gemaakt van een eigen Teams omgeving. Nadere informatie hierover wordt nog nader verstrekt. Dit kanaal is primair voor studenten onderling en is niet bedoeld als primair middel om docenten te contacten of om vragen te stellen.

Zoals eerder genoemd, is aanwezigheid op de projectdagen verplicht. Mocht je een keer echt niet kunnen komen (bijvoorbeeld in geval van ziekte), meld je dan vooraf af bij je Product Owner en coaches (per email) en bij je team.

3.5 Projectleiding

De organisatie van het Cyber Security project PAD Capture the Flag! is in handen van Tiemen Mink (t.mink@hva.nl). Hij is als deelprojectleider verantwoordelijk voor de planning van projectactiviteiten en de communicatie hierover naar de teams.

3.6 Weekplanning

Vanwege de beperkte tijd zijn vier sprints gedefinieerd. In elke sprint maakt een team Potentially Shippable Product en demonstreert dat aan (vertegenwoordigers van de) opdrachtgever.

w	Start week	Fase	Activiteiten (OnCampus en Project woensdag)
1	2 feb	Start/ Introductie	Kick-Off: Kennismaken opdrachtgever en opdracht, Thema's en Scrum
2	8 feb	Start	Thema kiezen, Backlog opstellen
3	15 feb	Sprint 1	Sprintplanning, Sprintgoal, Productplan, initialize (voorbereidingen)
4	22 feb	Vakantie	
5	1 mar	Sprint 1	Sprintdemo, Retrospective, Backlog grooming,
6	8 mar	Sprint 2	Sprintplanning, Sprintgoal
7	15 mar	Sprint 2	
8	22 mar	Sprint 2	Sprintdemo, Restrospective, Backloggrooming,
9	29 mar	Toetsweek	
10	5 apr	Toetsweek	
1	12 apr	Sprint 3	Sprintplanning, Sprintgoal
2	19 apr	Sprint 3	
3	26 apr	Sprint 3	Sprintdemo, Retrospective, Backlog grooming,
4	3 may	Vakantie	
5	10 may	Sprint 4	Sprintplanning, Sprintgoal
6	17 may	Sprint 4	
7	24 may	Sprint 4	Retrospective ,Product demo
8	31 may	Combitoets	Combitoets
9	7 jun	Toetsweek	Combitoets
10	14 jun	Toetsweek	Combitoets
11	21 jun	Toetsweek	
12	28 jun	Herkansing	Herkansing

4) Security

In dit project zijn we bezig met het creëren en aanvallen van systemen, applicaties en andere omgevingen die wetsbaarheden bevatten. Dat betekent een reëel gevaar voor andere omgevingen die hier slachtoffer van kunnen worden. Het is daarom zaak om een aantal strenge en bindende afspraken te maken.

4.1 Algemeen

Het gebruik van offensieve middelen en/of bedoelingen mag alleen plaatsvinden binnen het speciale CS-netwerk (zie 4.2) en alleen op speciaal hiervoor gemaakt Docker instances.
Ook mogen aanvallen alleen uitgevoerd worden met uitdrukkelijke toestemming van het aangevallen team gedurende een afgesproken tijdslot.
Als een student een aanval uitoefent buiten het speciale CS-netwerk, op andere omgevingen dan de hiervoor bedoelde Docker instances of zonder uitdrukkelijke toestemming van het aangevallen team, dan zal de student door de teamleiding uit het team worden gezet met dezelfde consequenties als bij de procedure team-uitzetting.

4.2 Netwerk

Tijdens het ontwikkelen/testen/aanvallen zijn studenten verplicht zijn om van het CS-netwerk gebruik te maken. Het CS-netwerk wordt extra gemonitord en kan bij calamiteiten afgesloten worden van de rest van het netwerk. Ongebruikelijk gedrag buiten het speciale CS-netwerk kan leiden tot uitsluiting van de student van het hele netwerk.
Wees ook in de thuissituatie erg voorzichtig. Je kunt met vulnerabilities bezig zijn die invloed hebben op je netwerk thuis maar ook op dat van je provider. Als je bijvoorbeeld een aanval uitvoert op je eigen Docker instances en je hebt je (virtueel) netwerk niet goed genoeg afgeschermd dan kan die aanval zomaar op je volledige netwerk en dat van je provider plaatsvinden. In het laatste geval loop je het risico op maatregelen van je provider.

Vooralsnog kun je op het CS-netwerk méér dan op de HvA- en Eduroam-netwerken. Op het CS-netwerk is horizontaal verkeer toegestaan. Dat houdt in dat studenten en medewerkers die op het CS-netwerk zijn ingelogd bij elkaars machines kunnen komen om die aan te vallen.
Inloggen op het wifi doe je met je HvA-ID en password. Het wifi SSID zendt alleen uit op de 2.4 Ghz band en niet op 5 Ghz. Dit om verstoringen in de standaard HvA-dienstverlening te beperken.
Het netwerk is beperkt van omvang, gebruik het alleen voor het project om kwetsbaarheden te ontwikkelen of om te sporen.

4.3 Security/Aanval procedure

Vooralsnog zijn alle securitymaatregelen procedureel.

- Elk studententeam moet een student aanwijzen die de CERT-rol (Computer Emergency Response Team) voor zijn rekening neemt.
- De student met de CERT-rol is ten alle tijde op de hoogte van aanvallen die uitgevoerd worden binnen zijn team en is aanspreekpunt voor de CERT-docenten.

- Voordat een aanval wordt uitgevoerd, is toestemming verkregen (mail) van de CERT-student van het aangevallen team.
- Wanneer een test of aanval misloopt of uit de hand loopt heeft de CERT-student een meldingsplicht en meldt het incident bij de CERT-docent.
- De CERT-docenten staan in direct contact met CERT-HvA zodat er eventueel direct ingegrepen kan worden.

5) Workshops

Gedurende het project worden workshops gegeven. De inhoud van deze workshops verschilt per deelproject. Heb je behoefte om meer te weten over een bepaald onderwerp, Geef dit dan aan bij één van de docenten.

5.1 Scrum

Werken met Scrum is een zeer belangrijk onderdeel van het project. Om die reden krijgen alle teams aan het begin van het project workshops over deze Agile methode. We zullen SCRUM workshops Just-In-Time aanbieden. Bekijk voor aanvang van het project alvast de Scrum Overview video en/of neem de Scrum Guide door. De kennis en tips van de resterende bronnen gebruik je gedurende het project. Op de DLO staat ook een document met de belangrijkste aspecten van SCRUM en hoe die in PAD gebruikt worden (PAD Scrum uitleg en Tips).

Documentatie

- Scrum Guide

Korte scrum video's

- Scrum Overview
- Scrum Increment
- Sprint Backlog
- Product Backlog
- Sprint Review
- Sprint Retrospective
- Scrum Sprint
- Scrum Values

Scrum tips

- Product Owner Tips
- Scrum Master Tips
- Daily Stand up tips

Agile bij een IT bedrijf

- Spotify Engineering Culture

Voor de liefhebbers; de diepte in

- Agile: Where are we at?

Agile Tools

- Trello — Digitaal scrumbord
- Spotify Retro Kit — Werkvormen voor retrospectives
- Miro — Digital whiteboard voor retrospectives (en brainstorms)
- Scrum Cheat Sheet — Infographic van alle scrum onderdelen

5.2 Overige workshop

Tijdens de eerste en tweede sprint zullen workshops worden georganiseerd over GITLab en over het werken met Docker containers.
Je krijgt ook SCRUM-workshops om die ontwikkelmethodiek je eigen te maken.
Naar behoefte kunnen ad-hoc workshops worden georganiseerd over andere relevante technische onderwerpen.

6) Toetsing en beoordeling

De eindbeoordeling bestaat uit de volgende drie onderdelen:

1. Productbeoordeling: Team cijfer voor 4 studiepunten
2. Proces beoordeling: Individueel cijfer voor 4 studiepunten
3. Methoden en technieken: Individueel cijfer voor 4 studiepunten

Alle onderdelen moeten met een voldoende worden afgerond om het project te halen.

6.1 Productbeoordeling

Productbeoordeling vindt plaats op basis van het opgeleverde product.

Beoordelingscriteria:

1. Bruikbaarheid: in hoeverre het systeem kan worden ingezet om de doelen van de opdrachtgever te bereiken.
2. Gebruiksgemak: in hoeverre de user interface van het systeem aansluit op en bruikbaar is voor doelgroep.
3. Omvang: De omvang van de opgeleverde functionaliteit, rekening houdend met de beschikbare ontwikkeltijd.
4. Complexiteit: De complexiteit van de opgeleverde functionaliteit, rekening houdend met de beschikbare ontwikkeltijd.
5. Kwaliteit: Het systeem is van dusdanige kwaliteit dat hij daadwerkelijk kan worden gebruikt door de doelgroep; het systeem bevat weinig of geen bugs.

Bij elk van deze criteria worden ook de eisen die de opdrachtgever stelt aan het gerealiseerde product meegenomen maar ook het correct opleveren van de deelproducten.

Normering

Per beoordelingscriteria kan je 2 punten verdienen met een maximum van 10 punten. Het totaal van deze punten vormt het cijfer.

Toetsmoment

De productbeoordeling vindt plaats aan het eind van de laatste sprint De herkansing van de productbeoordeling vindt plaats aan het eind van blok 4.

6.2 Combitoets

De procesbeoordeling en methoden en technieken worden tegelijk afgenomen in een combitoets. De totale tijd voor de combitoets is 2 lesuren (100 minuten) per team. De combitoets levert voor alle teamleden twee individuele cijfers op, één voor de procestoets en één voor de methoden & techniekentoets.

6.2.1 Procestoets

Als input voor deze toets dient de teamsite (zie paragraaf 7.3) welke jullie gedurende het project bijhouden.

Beoordelingscriteria:

1. Bijdrage: De mate waarin ieder teamlid een bijdrage heeft geleverd aan de verschillende producten binnen het project (alle producten die het team heeft opgeleverd).
2. Samenwerken: De mate waarin elk teamlid actief heeft bijgedragen aan het samenwerkingsproces. Onderdeel hiervan is de juiste en correcte uitvoering van SCRUM als team.
3. Zelfsturing: De mate waarin elk teamlid heeft laten zien te beschikken over zelfsturing.
   Denk hierbij aan zelfstandig werken, initiatief nemen, verantwoordelijkheid nemen voor eigen handelen en eigen gedrag, overzicht krijgen/behouden over de planning, voortgang bewaken van eigen en andermans taken, uit zichzelf hulp en feedback vragen of bieden aan andere teamleden. Onderdeel hiervan is de juiste en correcte uitvoering van SCRUM als lid van een team.
4. Ontwikkeling: De mate waarin elk teamlid zich heeft ontwikkeld.
   Weet elk teamlid hoe hij/zij wilde werken aan de verschillende hbo-competenties in het project en wat hij/zij wilde leren binnen het project? Heeft het teamlid daadwerkelijk die nieuwe kennis en vaardigheden opgedaan en bestaande kennis vergroot? Waren de leerdoelen van de individuele studenten bekend bij de anderen en werd daar rekening mee gehouden binnen het team? Hoe kritisch staat het teamlid tegenover zichzelf en is hij/zij zich bewust van eigen sterktes en zwaktes?

Bij elk van deze criteria wordt het correct hanteren van SCRUM als ontwikkelmethodiek voor het team in de beoordeling meegenomen.

Normering

Per beoordelingscriteria kan je 5 punten verdienen met een maximum van 20 punten. Het totaal van deze punten gedeeld door 2 vormt het cijfer.

6.2.2 Methoden & Techniekentoets

Voor deze beoordeling licht elk teamlid maximaal twee onderdelen van het IT-product toe waaraan hij/zij heeft gewerkt en trots op is. De leerroute van de student bepaalt welke onderdelen gebruikt mogen worden als bewijslast voor deze toets.
Van elke student wordt verwacht dat hij/zij zelfstandig een volledige vertical slice heeft gerealiseerd (van een challenge zowel de front-end, back-end, database van een zelfstandige challenge en de documentatie).

Beoordelingscriteria:

1. Kwantiteit van de door de student gemodelleerde, gerealiseerde en geteste onderdelen.
2. Kwaliteit van de door de student gemodelleerde, gerealiseerde en geteste onderdelen.
3. Complexiteit van de door de student gemodelleerde, gerealiseerde en geteste onderdelen.
4. Toelichting van de door de student gemodelleerde, gerealiseerde en geteste onderdelen.
5. Kennis en begrip van het product als geheel en van de samenhang van de verschillende onderdelen.

Normering
Per beoordelingscriteria kan je 5 punten verdienen met een maximum van 25 punten. Het totaal van deze punten gedeeld door 2,5 vormt het cijfer.

Scrumtoets (ondervoorbehoud)
In sprint 4 is het mogelijk een (facultatieve) scrum toets te maken. Als deze voldoende afgerond is, verdiend de student een bonuspunt op de M&T toets Dit geldt alleen voor studenten die zowel de scrum toets als de reguliere M&T toets voldoende afronden.
Of dit onderdeel wordt ingezet is afhankelijk van de mogelijkheid tot uitvoering ivm Corona maatregelen. Uiterlijk in sprint 3 zal hierover nadere informatie verstrekt worden.

6.2.3 Toetsmoment
De combitoets vindt plaats na de laatste sprint. De herkansing van de combitoets vindt plaats aan het eind van blok 4.

6.3 Procedure uitzetting uit de projectgroep

Bij het werken in projecten speelt samenwerken een cruciale rol. Wat doe je als een teamlid daartoe niet bereid is? Wat doe je als een teamlid consequent afspraken niet nakomt, zijn werk niet doet, te laat komt, enzovoorts? Natuurlijk kun je dat gedrag negeren, maar dat betekent dat de overige teamleden extra werk moeten doen en dat die leden dus eigenlijk niet ‘professioneel’ genoeg zijn om onacceptabel gedrag ter discussie te stellen. Zo’n team scoort niet hoog als het gaat om het proces. Is het dan handig om een niet-functionerend teamlid onmiddellijk, eventueel met harde hand, uit het team te verwijderen? Nee, ook dat is niet de meest voor de hand liggende oplossing. Iedereen heeft immers recht op een tweede kans. We hanteren de volgende regels:

1. Er vindt een crisisgesprek plaats tussen het niet-functionerend teamlid en de overige teamleden. De coach is bij dit crisisgesprek aanwezig. Tijdens dit gesprek wordt het functioneren van de groep en dat van individuele leden besproken. Aan het eind worden (nieuwe) afspraken gemaakt en volgt er een periode waarin het niet-functionerende lid moet bewijzen dat hij wél binnen het team kan functioneren. Met andere woorden: het teamlid moet een meetbare meerwaarde hebben en verantwoordelijk kunnen zijn voor het teamresultaat. De afspraken die in dit gesprek worden gemaakt worden schriftelijk vastgelegd.

2. Als deze periode is verstreken vindt er een nieuw gesprek plaats tussen de teamleden en de coach. Als volgens het team en de coach onvoldoende of geen voortgang is getoond dan wordt het betreffende lid uit het team gezet. Het behoort tot de mogelijkheden dat een team uiteindelijk verder gaat in het project met minder leden. In overleg met de projectleiding wordt dan eventueel een alternatief traject afgesproken. Het is natuurlijk niet de bedoeling dat een team van twee leden exact hetzelfde moet doen als een team van vijf studenten.

7) Deelproducten

Aan het eind van elke sprint levert het team een deel van het te realiseren IT-product op.
Daarnaast zijn er ook een aantal andere deelproducten welke opgeleverd moeten worden.

7.1 Leer- en samenwerkingsovereenkomst

In week 2 levert het team een leer- en samenwerkingsovereenkomst op. In het document leggen de teams vast wat zij doen om efficiënt en effectief met elkaar samen te werken en wat de leden zich als leerdoelen hebben gesteld.

De leer- en samenwerkingsovereenkomst bevat de volgende onderdelen:

1. De contactgegevens van alle teamleden
    a. Twee competentiegerichte individuele leerdoelen voor elk teamlid.
    Een SMART leerdoelen over professioneel vakmanschap
    b. Een SMART leerdoelen over communicatief vermogen
2. Duidelijke afspraken over hoe het team samenwerkt en leert met elkaar (inclusief regels over aan- en afwezigheid en onderlinge communicatie).
3. Regels over eventuele beëindiging van de samenwerking, zie hiervoor ook bijlage B.
4. Aan het eind de datum en de teamnamen met handtekeningen als bewijs dat alle teamleden akkoord gaan met de leer- en samenwerkingsovereenkomst.

7.2 Systeemdocumentatie

Gedurende het project wordt gewerkt aan systeemdocumentatie. Elke sprint wordt minimaal de gerealiseerde functionaliteit van dat moment opgeleverd. Enkele voorbeelden van relevante systeemdocumentatie zijn:

- Use-case scenario’s
- Use-case diagram
- Class diagram
- Code documentatie
- Design documentatie
- Interface documentatie

Denk eraan dat je Product na afloop van het project door anderen onderhouden moet kunnen worden.

7.3 Teamsite

Gedurende het project houdt elk projectteam een aantal zaken bij. In de DLO wordt daarvoor een aantal assignments aangemaakt.

De assignments bestaan uit drie onderdelen:

1. Producten

2. Retrospectives
3. Individuele reflecties

1. Producten
Alle producten die jullie gedurende het project opleveren (leer- en samenwerkingsovereenkomst, systeemdocumentatie, etc), moeten via de assignment CS Producten aanleveren. Dit is een groepsassignment. Zorg ervoor dat je alle bestanden van één versie in één zip bestand bij elkaar plaatst en voorzien is van relevante versie en datum aanduidingen zodat altijd inzichtelijk is wat de laatste versie is.

2. Retrospectives
Aan het einde van elke sprint houdt het team een retrospective. Hiervan moeten de volgende onderdelen worden gedocumenteerd:

- Een foto van alle post-its.
- De top 3 aandachtspunten die uit de retrospective volgen.
- Per aandachtspunt een SMART beschrijving hoe dit wordt aangepakt in de volgende sprint.
- Scrum artefacts van de sprint (Product backlog, Sprint goal, Sprint backlog, Definition of Done en Burndown chart).

Elke retrospective wordt in één PDF bestand opgeleverd in de groepsassignment “CS Retrospectives” en voorzien is van relevante retrospective, versie en datum aanduidingen zodat altijd inzichtelijk is wat de laatste versie is.

3. Individuele reflectie
De individuele reflectie bestaat per teamlid uit twee documenten:
Uiterlijk aan het einde van sprint 4 plaatst elk teamlid een individuele reflectie, waarin alle onderstaande onderwerpen zijn opgenomen:

- De onderdelen van het product waaraan hij/zij heeft bijgedragen. Als er is samengewerkt aan een onderdeel, geeft hij/zij aan met wie en hoe de werkverdeling was.
- Twee bijdragen waarop hij/zij trots is. Voorbeelden zijn een stuk code, een procesmodel of een UI-ontwerp. De user manual wordt hier bij buiten beschouwing gelaten.
- Reflectie op zijn/haar eigen handelen tijdens het project.
- Zijn/haar belangrijkste ontwikkeldoelen voor het tweede jaar.
- Feedback aan alle team leden.

Elke teamlid maakt van elk van deze producten één PDF bestand en levert deze op in de groepsassignment “CS Reflecties” en is voorzien is van de naam van het teamlid, versie en datum aanduidingen zodat altijd inzichtelijk is wat de laatste versie is.

8) Bijlagen

8.1 Bijlage A - Producteisen

8.1.1 Achtergrond

De opdrachtgever van dit project is Leerroute Cyber Security HBO-ICT, vertegenwoordigd door de leerroute manager Patrick Boom, geadviseerd door CS docenten uit de verdiepingsfase.
Gedurende de verdere ontwikkeling van de leerroute Cyber Security is meer en meer duidelijk geworden dat het noodzakelijk is dat er een omgeving nodig is waarin studenten en docenten Cyber Security threats kunnen ontwikkelen (training) en ontdekken. Een dergelijke omgeving traint de studenten in zowel het ontdekken, ontwikkelen en voorkomen van Cyber Security threats.
Voor een dergelijke omgeving is een Capture the Flag de meest aangewezen vorm. Door middel van gamificatie kan een studenten leren een threat te ontdekken en doorgronden en daarmee maatregelen bedenken om deze threats te voorkomen dan wel minimaliseren.
Tegelijkertijd is het mogelijk om een dergelijk CtF omgeving continue te voorzien van de laatst bekende threats en zo de studenten en docenten te trainen met de laatste inzichten.
Studenten en docenten kunnen (gecontroleerd) nieuwe challenges toevoegen aan deze omgeving.
Na enkele gesprekken is besloten om een dergelijke omgeving initieel te laten ontwikkelen door de studenten van CS jaar één in het project PAD.
De opdrachtgever zal gedurende het project nauw contact houden met de begeleiders van het project om ervoor te zorgen dat het project die producten oplevert die voor haar van belang zijn.
Tussentijds en aan het einde van het project zal een aantal van de studenten team producten aan de opdrachtgever gedemonstreerd worden en zal hierover feedback en een beoordeling geven.
Het is de bedoeling dat de producten aan het einde van het project door HBO-ICT-leerroute Cyber Security in gebruik genomen worden.

8.1.2 Opdracht

Elk team ontwikkeld voor de opdrachtgever één of meer modules voor een Capture the Flag (CTF).
Elk team kiest één Cyber threat thema uit de OWASP top 10, waarvoor zij de modules maken.

Een CTF is een wedstrijd waarbij een aantal deelnemers hun hacking skills kunnen laten zien in een gecontroleerde omgeving.

De organisatie (het development team) zet een omgeving op waarin zogenaamde flags gevonden kunnen worden, en de deelnemers moeten deze zien te verzamelen. Elke flag is punten waard, afhankelijk van de moeilijkheid, en aan het einde is het team met de meeste punten de winnaar.

Elk development team ontwikkeld meerdere challenges die bestaan uit een 3 tiers webapplicatie met daarin een aantal kwetsbaarheden.
De drie tiers zijn:
1. Frontend, de gebruiker interactie
2. Backend, de backendcode verwerkt input vanuit de frontend en slaat die op in een database
3. Database, in de database worden gegevens van de applicatie opgeslagen

Van elke student wordt verwacht dat hij/zij zelf een volledige vertical slice kan en heeft ontwikkeld.

Om zeker te weten dat de ontwikkelde CTF module aansluit bij de doelgroep is het ook belangrijk deze door potentiële gebruikers te laten testen en een rapportage op te stellen over hun bevindingen. Dit kan door het organiseren van een red-team / blue-team met hulp van medestudenten van andere teams.

Wat is nu precies een CTF-module? Een aantal voorbeelden is (soms na registratie) te vinden op:

- https://ctf.hacker101.com/ctf
- https://ctflearn.com/dashboard
- https://overthewire.org/wargames/

8.1.3 Product eisen

De volgende beschrijving geeft de door de opdrachtgever vastgesteld eisen weer. Uiteraard kunnen daar aanvullende zaken aan worden toegevoegd door de (team) product-owner. Afwijkingen van deze eisen kan alleen in overleg met de product-owner (docent).

Functionaliteit

Challenges
Elke Challenge is een drie-tiers webapplicatie en kan zelfstandig worden uitgevoerd na het starten op de door de CTF-framwork aangeven wijze. Voor elke challenge is het mogelijk om hints te verkrijgen, bij voorkeur gedoseerd.
Het is mogelijk dat er verschillende instances van een challenge tegelijkertijd actief zijn. Er kunnen meerdere gebruikers tegelijkertijd actief zijn.
Het is uiteraard niet mogelijk om op een andere wijze dan bedoeld in de challenge de flag te achterhalen.
Na verlaten, afsluiten of na een veilige tijd van inactiviteit wordt de instance opgeruimd.
De challenges voldoen aan de challenges beschrijving van het “**bsides-ctf-framework**”. Dit framework wordt gebruikt als basis voor alle challenges en de challenges sluiten hierop aan.

Kwaliteitscriteria en eisen
Alle gegevens zijn afgeschermd van andere gebruikers en alleen door het systeem te veranderen.
Elke verandering is zichtbaar in een audit-file.
Registratie, gebruik en opslag van gegevens is AVG-proof.
Er kunnen meerdere gebruikers tegelijkertijd een challenge uitvoeren.
Zij worden onderling niet beïnvloed.

De challenges worden in Docker gebouwd. Ze worden pas opgestart als een challenges daadwerkelijk wordt opgestart en ook na afloop weer opgeruimd.
Alle componenten worden in Gitlab opgenomen en het gehele ontwikkelproces is te volgen en herleiden in Gitlab (elke wijziging wordt hierin opgeslagen).

De challenges voldoen aan de challenges bechrijving van het "**bsides-ctf-framework**" zodat ze later ook in een eigen CTF omgeving geplaatst kunnen worden.

Frontend: Je maakt gebruik van eerder geleerde technieken als HTML/CSS, Python (WSGI) en Apache.
Programmeer conventies: HBO-ICT Code conventions for Python (zie bijlage B)
OS: Linus-based (X86 versie).
Backend: Python met Apache, gebruik van framwork en libraries noodzakelijk.
Database: MySQL of MariaDB, een ERD (zoals geleerd bij het vak Databases) is verplicht
Docker: Docker compose om een virtueel netwerkje te bouwen met daarin de drie tiers (elk van de drie tiers in een aparte Docker instance).
Versie management: HvA Gitlab (https://gitlab.fdmci.hva.nl/)

Omdat de CTF ook na het project gebruikt moet kunnen worden, is het belangrijk om goed te documenteren hoe de betreffende module beheerd en gebruikt kan worden, welke kwetsbaarheden aanwezig zijn en hoe deze kunnen worden gevonden.
Omdat elk development team onafhankelijk van de andere development teams werkt maar na afloop de verschillende challenges in één CtF omgeving geplaatst moeten kunnen worden is het belangrijk dat de afhankelijkheid tussen de front-end en de challenges minimaal is, duidelijk en volledig gedocumenteerd is en gemakkelijk te implementeren is in een ander front-end.
Onderdeel van de documentatie is een architectuurplaat.

Challenges
Voor de challenges zijn er x thema's waaruit de teams kunnen kiezen.
Elk team kiest één thema waarvoor challenges worden gemaakt. Alleen in uitzonderlijke gevallen (er zijn geen mogelijkheden maar om een challenge te maken) en in overleg met de opdrachtgever mogen challenges voor twee thema's gemaakt worden.
Elk team maakt minimaal drie challenges, één in elk van de moeilijkheid categorieën 'easy'(beginner), 'to do'(gemiddeld) en 'tough'(moeilijk). Deze drie challenges zijn geen voortzetting van elkaar maar zijn verschillend binnen een thema.

De thema's (OWASP top 10):

- A1:2017-Injection
- A2:2017-Broken Authentication
- A3:2017-Sensitive Data Exposure
- A4:2017-XML External Entities (XXE)
- A5:2017-Broken Access Control (*
- A6:2017-Security Misconfiguration
- A7:2017-Cross Side Scripting (*
- A8:2017-Insecure Deserialization
- A9:2017-Using Components with Known Vulnerabilities
- A10:2017-Insufficient Logging & Monitoring

8.2 Bijlage B – HBO-ICT Code Conventions voor Python

HBO-ICT Code Conventions voor Python

Bij het programmeren is het belangrijk, dat je broncode leesbaar en begrijpelijk is. Als we ons allemaal aan dezelfde afspraken houden, wordt het makkelijker code te schrijven en onderhouden, en ook om daarbij als team samen te werken. We maken daarom gebruik van “Code Conventions”.
De volgende afspraken zijn “verplicht” en worden ook meegewogen in de beoordeling van je werk:

- Naam (voornaam en achternaam) van de auteur en het doel van het programma staan als commentaar bovenaan het programma.
- De code wordt voorzien van voldoende commentaar, om de werking van de code te verduidelijken. Commentaar moet het waarom toelichten en niet herhalen wat er al staat
    - Dus niet: `index+=1 # verhoog index met 1`
    - Maar: `index+=1 # verwijs naar volgende student in de rij`
- Functies worden voorafgegaan door commentaar waarin het doel van de functie, de mee te geven argumenten en de return value beschreven staan
- Variabelen, functies en methodes hebben duidelijke namen, waaruit het bedoelde gebruik duidelijk blijkt.
- Voor de naamgeving wordt gebruik gemaakt van lowerCamelCase (beginnen met kleine letter, en elk volgend woord direct daaraan vast met hoofdletter) of van snake_case (allemaal kleine letters, met underscore tussen de woorden).
- Gebruik voor de naamgeving wel consequent lowerCamelCase of snake_case. Gebruik ook consequent de Nederlandse, of de Engelse taal. Dit geldt ook voor je commentaar.
- Regels niet langer dan 120 tekens, liefst korter dan 100
- Consequente indentatie (inspringing) van 4 spaties

Buiten de hierboven genoemde punten, zijn er nog een aantal andere Code Conventions, die je helpen begrijpelijke code te schrijven. Maak hier gebruik van!

- Importeer modules bovenin je programma, direct na de regels commentaar met het doel van het programma en je naam. Importeer geen modules, die je niet gebruikt.

- Verwijder statements die weg kunnen en maak je programma “schoon”. Dit geldt ook voor code die “commented-out” is.
- Vermijd Magic Numbers: getallen die opeens in programmacode voorkomen en waarvan de functie niet duidelijk is.
    - Dus niet: `bedragInEuro = aantalBitcoin * 7432`

      Maar: `bedragInEuro = aantalBitcoin * koersBitcoinInEuro`
- Organiseer je programma in blokken programma-code. Een blok is een aantal regels code die functioneel bij elkaar horen. Scheidt de blokken met witregels. (Gebruik geen witregel na elke losse regel code.)