

Opdracht

Ontwikkelen app BTP (Boomkwekerij Taak Planner)

1 Kadering

Tijdens het olod Applied Software Project ga je in teamverband een project uitwerken voor een gegeven behoefteanalyse.

De aanpak van het project wordt opgedeeld in 3 fases:

- Fase 1: Functionele analyse en studie,
- Fase 2: Ontwikkeling (inclusief testen en documenteren),
- Fase 3: Oplevering en verdediging.

Je werkt 16 uren per week waarvan 8 uren verplicht on campus. Het tijdstip en lokaal kan je steeds terugvinden in Webuntis.

2 Fase 1

2.1 Context

In deze fase ga je de behoefteanalyse doornemen en ga je van start met de blueprint op te stellen. Je voorziet daarbij minstens volgende documentatie:

- User stories,
- Activity diagram(men) voor minstens 1 user story per teamlid,
- Sequence diagram(men) voor minstens 1 user story per teamlid,
- Scherm design(s) voor minstens 1 user story per teamlid,
- Acceptatiecriteria voor minstens 1 user story per teamlid,
- Entity Relation Diagram (ERD) van het ganse project door het team.

Elk teamlid werkt daarbij minstens 1 volledige user story (inclusief diagrammen en scherm designs) zelfstandig uit. Je bundelt je user stories tot een product backlog die als startpunt zal dienen voor de ontwikkelingsfase.

Naast de analyse ga je je technische skills verder verdiepen in zelfstudie a.d.h.v. een aantal lessen. We adviseren om deze studie op te delen in 2 weken met telkens 4u per week.

2.2 Te gebruiken tools

- GitLab/GitHub, om alle documentatie onder te brengen en je product backlog.
- MS Word, voor het uitschrijven van de user stories en acceptatiecriteria.
- Draw.io, voor het uitwerken van de diverse diagrammen.
- Adobe XD of gelijkwaardig, voor het uitwerken van de scherm designs.

2.3 Deliverables

- Per user story gebundeld in een map en toegevoegd aan je repository:
 - De beschrijving in een Word document,
 - Acceptatiecriteria in een Word document,
 - Activity diagram(men) bij voorkeur in 1 bestand gebundeld,
 - Sequence diagram(men) bij voorkeur in 1 bestand gebundeld,
 - Scherm designs als afbeelding.
- ERD.

3 Fase 2

3.1 Context

In de 2de fase werk je het project full-stack uit op basis van je eigen blueprint en doet dit m.b.v. de Scrum methodologie.

In je GitHub team repository wordt daartoe een project gekoppeld waarin je je team Scrum board kan terugvinden. Het board is onderverdeeld in 5 buckets waarbij elke bucket een status representeert:

- Product backlog
- To do
- In progress
- In test
- Done

De bucket “Product backlog” moet initieel alle user stories van je project bevatten.

Naast je Scrum board in GitHub is er ook een Excel werkmap toegevoegd aan het privé kanaal van je team in MS Teams (tab Bestanden). Hierin zal je bij elke sprint het sprintverslag moeten inwerken.

De ontwikkeling en testing van een user story wordt altijd volledig door één teamlid uitgewerkt. Op die manier bouw je ervaring op in back-end en front-end ontwikkeling. Het is uiteraard niet verboden om je user story extra te laten testen door een ander teamlid.

3.2 Praktisch

Voorafgaand aan de eerste sprint planning:

1. Je zorgt ervoor dat je alle user stories in je Scrum board onder de bucket “Product backlog” plaatst. Je doet dit door in deze bucket een issue aan te maken voor elke user story.
2. Je kent aan elke user story een prioriteit toe (veld Priority in de issue) en sorteert je Product backlog op basis van deze prioriteit (belangrijkste eerst).

Per sprint:

1. Sprintplanning
 - a. Je plant je sprint in, gebruik makend van de planning poker methode, en kent daarbij story points toe aan de user stories die je in de sprint zal opnemen. Je sprint eindigt in principe altijd op woensdagavond voorafgaand aan de contacturen op donderdag.
 - b. Je verplaatst deze user stories naar de bucket “To do”. De bucket “To do” wordt dus als het ware je sprint backlog. User stories die je vanuit een voorgaande

sprint opnieuw meeneemt blijven in de status staan waarin ze zich bevonden aan het einde van de voorgaande sprint.

- c. Je past de Milestone van de user stories die je in de sprint zal opnemen aan naar de juiste sprint. Wanneer je een user story uit een vorige sprint opnieuw mee opneemt pas je ook daar de milestone aan naar de nieuwe sprint.
- d. Je neemt deze user stories onmiddellijk mee over in de Excel werkmap van het sprintverslag in je Teams kanaal.

2. Tijdens de sprint

- a. Je onderhoudt je Scrum board consequent en verplaatst elke user story naar de juiste status wanneer deze veranderd.
- b. Je houdt consequent je gewerkte tijd bij per user story in de Excel werkmap van het sprintverslag in je Teams kanaal.
- c. **Je werkt in een user story volledig zelfstandig uit (back-end en front-end).** Enkel tijdens de eerste sprint mag je nog samenwerken indien dat nodig is.
- d. Je houdt hierbij rekening met de technische vereisten zoals aangegeven in het gelijknamige deel van deze opdracht.
- e. Elke user story wordt functioneel getest en pas wanneer minstens aan alle acceptatiecriteria is voldaan mag de user story als afgewerkt beschouwd worden en verplaatst worden naar bucket "Done".

3. Aan het einde van een sprint

- a. Je vult in de Excel werkmap van het sprintverslag de ontbrekende data bij aan. De instructies hiervoor kan je terugvinden in de Excel werkmap.

Dit moet ten laatste aan het einde van elke sprint om 19:00u gebeurd zijn.

3.3 Te gebruiken tools

- GitHub, om je codebase onder te brengen en je product/sprintbacklog en Scrum board bij te houden.
- Visual Studio, om je code in uit te werken.
- MS Teams en MS Excel om je sprintverslag in te vullen.

3.4 Deliverables

- Per sprint
 - Een geactualiseerd Scrum board in GitHub met daarin de user stories die je opgenomen hebt tijdens de sprint, inclusief de story points en prioriteit.
 - Je codebase in GitHub.
 - Je team sprintverslag in de Excel werkmap op MS Teams. Hou daarbij rekening met de eerder aangegeven deadline.

4 Planning

TIP: Gebruik je Outlook kalender om deadlines in te plannen zodat je reminders krijgt!

Jaar	Week	Datum	Focus	Deadline
2022	38	22-23/09/2022	Fase 1 - Analyse	
	39	29-30/09/2022	Fase 1 - Analyse	
	40	05-06/10/2022	Fase 1 - Analyse & studie	
	41	13-14/10/2022	Fase 1 - Analyse & studie	
	42	17/10/2022	Fase 2 - Ontwikkeling	Sprint 1
	43	26/10/2022	Fase 2 - Ontwikkeling	Einde sprint 1 Oplevering blueprint
	44		Herfstvakantie	
	45	27/10/2022	Fase 2 - Ontwikkeling	Start sprint 2
	46	16/11/2022	Fase 2 - Ontwikkeling	Einde sprint 2
	47	17/11/2022	Fase 2 - Ontwikkeling	Start sprint 3
	48	30/11/2022	Fase 2 - Ontwikkeling	Einde sprint 3
	49	1/12/2022	Fase 2 - Ontwikkeling	Start sprint 4
	50	14/12/2022	Fase 2 - Ontwikkeling	Einde sprint 4
	51	23/12/2022	Fase 2 - Ontwikkeling	Code freeze!
	52		Kerstvakantie	
2023	1		Kerstvakantie	
	2		Fase 3 - Oplevering	Presentatie

5 Technische vereisten

Je houdt bij de uitwerking van je project rekening met volgende technische vereisten:

- Database
 - SQL Server
- Technologie
 - .NET 6.0
 - Back-end minimaal met EF Core
 - Front-end admin gedeelte met ASP.NET MVC
 - Front-end medewerker app met Ionic/Angular
 - Authenticatie met Auth0
- Overige technische vereisten
 - Clean code architectuur toepassen
 - Minimaal 2 unit testen per user story
 - Minstens 1 externe API aanspreken
 - Minimaal 10 business rules uitwerken waarvan minstens 5 rules uit de lijst komen die is aangegeven onder het hoofdstuk Behoeftanalyse. Je mag dus ook zelf business rules definiëren die niet in de lijst staan.

6 Behoeftanalyse

6.1 Contextschets

Voor een boomkwekerij moet een nieuwe app ontwikkeld worden. De app moet het mogelijk maken voor medewerkers om met een smartphone (of andere device) een takenlijst te raadplegen die aangeeft welke taken de medewerker die dag moet uitvoeren.

6.2 Behoeften

De boomkwekerij omvat meerdere kweeksites in Vlaanderen. Van elke kweeksite willen we naast de adresgegevens ook een plattegrond (afbeelding) bij kunnen houden. Deze plattegrond geeft altijd de onderverdeling in zones weer. Een zone is een deel van de kweeksite waarop een bepaalde boomsoort gekweekt wordt. Elke zone heeft bepaalde afmetingen die niet noodzakelijk hetzelfde zijn.

Al deze informatie over kweeksites moet toegankelijk gesteld worden vanuit een admin gedeelte van de applicatie.

Aangezien we bomen kweken moeten we ook informatie over bomen kunnen beheren in het admin gedeelte van de applicatie. Deze informatie omvat onder andere foto's (ter herkenning), de zone(s) op de kweeksite(s) waar de bomen gekweekt worden, de onderhoudsinstructies in pdf-formaat en een QR-code. Deze QR-code wordt dan bij de zone waarin de bomen gekweekt worden geplaatst. Wanneer medewerkers op basis van hun takenlijst de bomen moeten verzorgen kunnen ze de QR-code inscannen om het juiste onderhoudsplan te openen. Dit onderhoudsplan verschilt immers van seizoen tot seizoen.

De takenlijst van een medewerker wordt opgesteld door een verantwoordelijke die dat ineens voor alle dagen van de week doet. Elke dag resulteert dus in een specifieke planning per medewerker.

Wanneer een medewerker een taak start moet hij op een eenvoudige manier kunnen aangeven dat hij aan de taak begint. Wanneer de medewerker klaar is moet hij op een gelijkaardige manier kunnen aangeven dat hij/zij klaar is met de taak.

In een later stadium kunnen we dan via deze gegevens de efficiëntie van de medewerkers gaan analyseren. Het raadplegen van de takenlijst, starten en stoppen van taken en het raadplegen van onderhoudsplannen moet dus via een aparte app op de smartphone kunnen. Het spreekt voor zich dat een medewerker zich moet aanmelden bij de app zodat niet gemachtigde personen geen toegang hebben.

6.3 Business rules

Eenvoudige business rules:

1. Een medewerker mag maximaal in 2 sites tewerkgesteld worden.
2. Een medewerker werkt 1 week steeds in dezelfde site.
3. Een medewerker mag maximaal 5 dagen per week werken (ma-zo).
4. Een medewerker mag maximaal 4 taken per dag toegekend krijgen.
5. Een zone mag per dag maximaal voor 1 medewerker ingepland worden.
6. Een zone moet dagelijks minstens 1x ingepland worden.
7. Een boomsoort mag maximaal in 3 zones worden ondergebracht binnen 1 site.
8. Een taak kan slechts 1x gestart worden.
9. Een taak kan slechts 1x gestopt worden.
10. Een taak die afgehandeld werd kan niet opnieuw gestart worden.
11. Taken moeten na elkaar afgewerkt worden (m.a.w. voor een medewerker kan er max. 1 taak tegelijk "in progress" staan).
12. Taken hebben een bepaalde volgorde van inplanning en afwerking.

Moeilijkere business rules:

1. Een medewerker kan deeltijds in 1 en deeltijds in een andere site werken binnen 1 week.
2. Een medewerker kan deeltijds tewerkgesteld zijn.
3. Een taak heeft een bepaalde tijd nodig en een medewerker mag max. voor 8u taken ingepland worden.
4. Een zone moet ingepland worden volgens de specificaties van het onderhoud, 1x per dag, om de 2 dagen,..
5. Om variatie te geven mogen medewerkers niet 2 dagen na elkaar dezelfde zones toegekend krijgen.