# Functionele & non-functionele requirements

## Functionele Requirements

1. **Projectmanagement volgens Scrum**: Ondersteuning voor het bijhouden van requirements in de vorm van een product backlog, het opsplitsen van backlog items in activiteiten, toewijzen van backlog items aan developers binnen sprints, en de mogelijkheid tot discussie via berichten.
2. **Software Configuration Management (SCM)**: Integratie met bestaande version control systemen zoals Git of Subversion om de ontwikkelde code van projecten te beheren.
3. **DevOps en Development Pipelines**: Ondersteuning voor het opzetten van development pipelines met acties voor het ophalen van source code, installeren van packages, builden en linken van software, uitvoeren van tests, analyseren van code, deployment en diverse utilities.
4. **Discussieforum**: Een forum waarop discussies over backlog items gevoerd kunnen worden, met de mogelijkheid om nieuwe berichten en discussiethreads te starten of te reageren op bestaande threads.
5. **Rapportage voor elke Sprint**: Generatie van rapporten met informatie over de teamsamenstelling, burndownchart, aantal effort-punten per developer, etc., inclusief de mogelijkheid om headers en footers toe te passen voor een professionele uitstraling.
6. **Rollen en Toegangsbeheer**: Verschillende rollen binnen het systeem, waaronder developers, een scrum master per sprint, en een product owner op projectniveau.
7. **Notificaties**: Automatische notificaties naar relevante stakeholders (zoals testers, scrum master, product owner) wanneer bepaalde acties plaatsvinden, zoals het veranderen van de status van backlog items, starten van sprints, en resultaten van development pipelines.
8. **Sprint Management**: Beheren van sprints inclusief het toewijzen van backlog items aan sprints, het doorlopen van verschillende fasen binnen een sprint (zoals todo, doing, ready for testing, etc.), en het afhandelen van sprints afhankelijk van de uitkomst (afsluiten, reviewen, of deployment initiëren).
9. **Integratie met externe communicatiemiddelen**: Ondersteuning voor het versturen van notificaties en communicatie via verschillende media (zoals e-mail, Slack, etc.) om de flexibiliteit in communicatie tussen teamleden te vergroten.
10. **Discussie en Documentatie**: Mogelijkheid voor gebruikers om discussies te voeren en documentatie bij te houden binnen het systeem, gerelateerd aan specifieke backlog items of algemenere projectthema's.
11. **Versiebeheer en Code Archief:** Mogelijkheid om wijzigingen in code te beheren door middel van integratie met SCM-systemen, inclusief basisfuncties zoals commit, branch, etc., en koppeling van code aan specifieke backlog items of sprints.
12. **Deployment en Release Management**: Mogelijkheid om software te deployen als onderdeel van de development pipeline, inclusief ondersteuning voor verschillende soorten deployments (zoals naar testomgevingen of productie).

## Non-Functionele Requirements

1. **Kwaliteit van de Code**: De code dient beoordeeld te worden met een Quality Gate label A in SonarCloud, wat wijst op hoge codekwaliteit.
2. **Onderhoudbaarheid en Testbaarheid**: Implementatie van unit tests voor de logica in het domein, met specifieke aandacht voor path coverage voor onderdelen met een complexere structuur. Het gebruik van stub-implementaties en het mocken van externe afhankelijkheden is vereist.
3. **Automatisering in DevOps-omgeving**: Gebruikmaken van een DevOps-omgeving naar keuze voor source-code-management en het automatiseren van development pipelines, inclusief bouwen, testen, en statische code analyse.
4. **Flexibiliteit en Uitbreidbaarheid**: Het ontwerp moet ruimte bieden voor variatie en de mogelijkheid ondersteunen om meer varianten van concepten (bijv. verschillende versies van SCM of DevOps acties) te integreren, met behoud van de ondersteuning voor de kernfunctionaliteiten.
5. **Schaalbaarheid en Prestatie**: Het systeem moet in staat zijn om te schalen om verschillende groottes van teams en projecten te ondersteunen, zonder significante verlies van prestatie.
6. **Veiligheid**: Bescherming van gevoelige informatie en code binnen het systeem, inclusief passende authenticatie- en autorisatiemechanismen om toegang te controleren.
7. **Compatibiliteit en Integratie**: Het systeem moet compatibel zijn met gangbare tools en platforms gebruikt in softwareontwikkeling en DevOps-praktijken, en gemakkelijk integreren met externe systemen en services.

# Design Patterns

## Creational Patterns

**Factory Pattern**: Deze kunnen worden gebruikt voor het creëren van objecten zonder dat je de exacte klassen van de objecten die gecreëerd moeten worden, specificeert. Dit is bijzonder handig voor het aanmaken van verschillende notificatiehandlers (e-mail, Slack, SMS) of verschillende integraties met externe systemen (verschillende SCM-systemen of DevOps-tools), waardoor het systeem flexibel en uitbreidbaar blijft.

## Structural patterns

1. **Adapter Pattern**: Maakt interactie mogelijk tussen onverenigbare interfaces, zoals bij het aansluiten van verschillende SCM-systemen of DevOps-tools die niet direct compatibel zijn met de interface van uw systeem.
2. **Composite Pattern**: Helpt bij het beheren van een groep objecten op dezelfde manier als een enkel object. Dit is nuttig voor het modelleren van de hiërarchische structuur van projecten, waarbij projecten, sprints, en backlog items als een eenheid of afzonderlijk kunnen worden behandeld.

## Behavioural Pattern

1. **Strategy Pattern**: Biedt de mogelijkheid om het algoritme dat door een object wordt gebruikt dynamisch te veranderen. Het kan bijvoorbeeld worden toegepast in het notificatiesysteem om flexibel het kanaal te wisselen waarmee meldingen worden verzonden.
2. **Observer Pattern**: Staat centraal in het observeren van en reageren op de staat van andere objecten, zoals het versturen van automatische notificaties bij statuswijzigingen van backlog items of sprints.
3. **State Pattern**: Hiermee kan het gedrag van een object veranderen wanneer zijn interne staat verandert. Dit is bijzonder geschikt voor het beheren van de levenscyclus en statussen van sprints en backlog items binnen het projectmanagement.