**Dependency-Management**

**Wat Is een Dependency-Management?**

Dependency management verwijst naar het proces van het beheren en beheersen van de afhankelijkheden van een softwareproject.   
Afhankelijkheden zijn externe bibliotheken, frameworks, modules of andere softwarecomponenten die nodig zijn voor het correct functioneren van de applicatie. Deze afhankelijkheden worden vaak gebruikt om herbruikbare functionaliteit toe te voegen.

**Composer** wordt door de meeste moderne PHP frameworks gebruikt als dependency manager. De 'composer.json' file wordt gebruikt om packages/libraries toe te voegen aan een project, waarin de namen en de versies van de binnen te halen packages/libraries worden bijgehouden.

Enkele belangrijke kenmerken van Composer zijn:

1. **Pakketbeheer**: Composer stelt ontwikkelaars in staat om bibliotheken van externe bronnen, zoals Packagist, toe te voegen aan hun PHP-projecten.
2. **Autoload**-**functionaliteit**: Composer genereert automatisch een autoload-bestand voor het project, waardoor het eenvoudig wordt om klassen en functies van externe pakketten te gebruiken zonder handmatig require- of include-statements toe te voegen aan het project.
3. **Versiebeheer**: Composer ondersteunt versiebeheer, waardoor ontwikkelaars specifieke versies van pakketten kunnen specificeren en beheren in hun projecten.
4. **Gedefinieerde** **configuratiebestanden**: Composer maakt gebruik van een JSON-configuratiebestand genaamd `composer.json`, waarin ontwikkelaars de afhankelijkheden (bijv. bibliotheken) van hun projecten definiëren.

`composer.json` en `composer.lock` zijn twee belangrijke bestanden die worden gebruikt bij het werken met Composer voor dependency management in PHP-projecten.

1. composer.json:

- `composer.json` is een JSON-configuratiebestand dat wordt gebruikt om de afhankelijkheden en configuratie van een PHP-project te definiëren.

- In dit bestand worden de externe bibliotheken en frameworks gespecificeerd die het project nodig heeft, samen met hun versienummers en andere vereisten.

- Het bevat ook andere projectconfiguraties, zoals autoload-instellingen, scripts en vereisten.

- Ontwikkelaars bewerken dit bestand om nieuwe afhankelijkheden toe te voegen, bestaande afhankelijkheden bij te werken of projectconfiguraties aan te passen.

2. composer.lock:

- `composer.lock` is een ander JSON-bestand dat wordt gegenereerd door Composer nadat de afhankelijkheden van een project zijn geïnstalleerd.

- Dit bestand bevat een nauwkeurige lijst van alle geïnstalleerde afhankelijkheden, inclusief de exacte versienummers en de versienummers van hun eigen afhankelijkheden.

- Het wordt gebruikt om de exacte staat van de afhankelijkheden vast te leggen op een bepaald moment in de tijd, inclusief alle transitive dependencies (afhankelijkheden van afhankelijkheden).

- `composer.lock` wordt meestal gebruikt in productieomgevingen om ervoor te zorgen dat de exacte versies van de afhankelijkheden consistent blijven tussen verschillende omgevingen.

- Dit bestand wordt automatisch bijgewerkt door Composer wanneer afhankelijkheden worden geïnstalleerd of bijgewerkt, maar het wordt niet direct bewerkt door ontwikkelaars.