# Gevonden problemen

In dit hoofdstuk zal er een overzicht komen van alle problemen. Dit zal per class worden gedaan en genummerd zodat dit overzichtelijk is.

### 1 Accessor;

* + 1.1 De constructor staat niet direct onder de fields.

### 2 BitmapItem.java;

* + 2.1 Lege extra constructor;
  + 2.2 Try catch in constructor.

### 3 DemoPresentation;

* + 3.1 Alle slides staan in 1 methode.

### 4 JabberPoint

* + 4.1 Try catch in constructor.

### 5 MenuController;

* + 5.1 Onoverzichtelijke constructor.

### 6 Slide;

* + 6.1 draw is onoverzichtelijk.

### 7 SlideItem;

* + 7.1 Lege extra constructor;

### 8 SideViewerComponent;

* + 8.2 paintComponent voert al dingen uit voordat er wordt gecheckt als de slide wel bestaat.

### 9 SlideViewerFrame;

* + 9.2 setupWindow is onoverzichtelijk.

### 10 Style;

* + 10.1 De constructor staat niet direct onder de fields.

### 11 TextItem;

* + 11.1 Lege extra constructor;

### 12 XMLAccesor;

* + 12.1 loadFile onoverzichtelijk door te veel inentingen;
  + 12.2 saveFile onoverzichtelijk door te veel inentingen.

### 13 Algemeen.

* + 13.1 Niet overal wordt this gebruikt waar dit nodig is;
  + 13.2 Meerdere fields wordt meteen al op ingevuld en later in de constuctor overschreven;
  + 13.3 Niet alle fields hebben modifiers of ze staan in de verkeerde volgorde;
  + 13.4 Alle classes zitten in 1 map;
  + 13.5 De grotere methodes zijn onduidelijk aangezien alles direct onder elkaar staan;
  + 13.6 In een aantal classes staat de ongebruikte fields serialVersionUID.

# Probleemoplossing

In dit hoofdstuk zal er voor de gevonden problemen een oplossing worden voorgesteld met en toelichting waarom de oplossing wordt toegepast.

### 1 Accessor;

* + 1.1

De constructor zal onder de fields worden gezet. Dit omdat het zo overzichtelijker is wat de constructor is en dat dit de standaardregels zijn.

### 2 BitmapItem.java;

* + 2.1

De extra constructor zal worden verwijderd. Dit omdat deze niks doet en daardoor overbodig is.

* + 2.2

De try catch zal in zijn eigen methode worden geplaatst. Dit omdat het op deze manier overzichtelijker is.

Verder zal dit ervoor zorgen dat als de class in de toekomst wordt aangepast dit overzichtelijk blijft.

### 3 DemoPresentation;

* + 3.1

Alle slides worden in hun eigen methode gezet. Dit omdat het zo overzichtelijker is wat er gebeurd en alles mooi geordend staat.

### 4 JabberPoint

* + 4.1

De try catch zal in zijn eigen methode worden geplaatst. Dit omdat het op deze manier overzichtelijker is.

Verder zal dit ervoor zorgen dat als de class in de toekomst wordt aangepast dit overzichtelijk blijft.

### 5 MenuController;

* + 5.1

De code zal worden onderverdeeld in hun eigen methodes. Dit omdat het op deze manier overzichtelijker is.

### 6 Slide;

* + 6.1

White lines plaatsen. Hierdoor zal het overzichtelijker worden doordat het niet allemaal direct onder elkaar staat.

### 7 SlideItem;

* + 7.1

De extra constructor zal worden verwijderd. Dit omdat deze niks doet en daardoor overbodig is.

### 8 SideViewerComponent;

* + 8.1

De if check wordt boven aan de methode geplaatst. Dit omdat hij dan eerst checkt als het wel mogelijk is.

### 9 SlideViewerFrame;

* + 9.1

White lines plaatsen. Hierdoor zal het overzichtelijker worden doordat het niet allemaal direct onder elkaar staat.

### 10 Style;

* + 10.1

De constructor zal onder de fields worden gezet. Dit omdat het zo overzichtelijker is wat de constructor is en dat dit de standaardregels zijn.

### 11 TextItem;

11.1

De extra constructor zal worden verwijderd. Dit omdat deze niks doet en daardoor overbodig is.

### 12 XMLAccesor;

* + 12.1

De methode zal worden gesplitst over meerdere classes. Dit om meer duidelijkheid te geven wat er gebeurt binnen de for loops.

* + 12.2

De methode zal worden gesplitst over meerdere classes. Dit om meer duidelijkheid te geven wat er gebeurt binnen de for loops.

### 13 Algemeen.

* + 13.1

This plaatsen bij alle niet static fields. Dit omdat dit duidelijkheid geeft welke fields van de class zijn en welke locale fields zijn.

* + 13.2

De field gaan invullen in de constructor. Dit is standaard en door alle aan de constructor toe te voegen krijg je dat alles op dezelfde manier wordt gedaan.

* + 13.3

Aan de fields zonder modifiers een toevoegen en zorgen dat alles gelijk is. Dit omdat je ze krijgt dat alles op dezelfde manie wordt gedaan.

* + 13.4

De classes onderverdelen in meerdere mappen per category. Dit zorgt ervoor dat er meer duidelijkheid komt in de mappenstructuur.

* + 13.5

Whitelines toevoegen, om zo ervoor te zorgen dat er een beter overzicht komt wat er gebeurd in de methodes door alles onderverdelen met de whitelines.