7-6-2017

Patrick van Batenburg,

Steven Logghe

Samen Sterk

Documentatie van de aanpassingen

Inhoudsopgave

[Inleiding 2](#_Toc485245946)

[Aanpassingen 2](#_Toc485245947)

[Controllers.AppointmentController 2](#_Toc485245948)

[Controllers.GradeController 2](#_Toc485245949)

[Controllers.RepeatingTaskController 3](#_Toc485245950)

[Controllers.SubjectController 3](#_Toc485245951)

[Controllers.TaskController 3](#_Toc485245952)

[Controllers.UserController 4](#_Toc485245953)

[Views.DeleteUsers 4](#_Toc485245954)

[Views.EditSubject 4](#_Toc485245955)

[Views. EditTask 4](#_Toc485245956)

[Views.Login 4](#_Toc485245957)

[Views.Register 4](#_Toc485245958)

[Views.Shedule 4](#_Toc485245959)

[Revisies 5](#_Toc485245960)

# Inleiding

In dit document worden de aanpassingen naar aanleiding van de testbevindingen gedocumenteerd.

# Aanpassingen

## Controllers.AppointmentController

In de methode Details(uint) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Create(Appointment) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Edit(Appointment) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Delete(Appointment) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

## Controllers.GradeController

In de methode Details(uint) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Create(Grade) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Edit(Grade) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Delete(Grade) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

## Controllers.RepeatingTaskController

In de methode Details(uint) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Create(RepeatingTask) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Exceeds(RepeatingTask) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Edit(RepeatingTask) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Delete(RepeatingTask) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen. Verder moest er gecheckt worden op de parameter of dat deze wel geïnitialiseerd is

## Controllers.SubjectController

In de methode Details(uint) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Create(Subject) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Edit(Subject) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Delete(Subject) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

## Controllers.TaskController

In de methode Details(uint) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Create(Task) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Exceeds(Task) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Edit(Task) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Delete(Task) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen. Verder moest er gecheckt worden op de parameter of dat deze wel geïnitialiseerd is

## Controllers.UserController

In de methode Details(User) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Register(User) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Edit(User) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode Delete(User) moest er een try-catch statement komen voor de connectie met de database, om zo onverwachte database fouten te kunnen afvangen.

In de methode LogOff(Type) moest er gecheckt worden of dat de parameter niet van type null was.

## Views.DeleteUsers

Nadat fouten in de UserController opgelost waren, moest de waarde waarop gecheckt wordt of dat iets juist of onjuist is aangepast worden.

## Views.EditSubject

Nadat fouten in de SubjectController en GradeController opgelost waren, moest de waarde waarop gecheckt wordt of dat iets juist of onjuist is aangepast worden.

## Views. EditTask

Nadat fouten in de TaskController en RepeatingTaskController opgelost waren, moest de waarde waarop gecheckt wordt of dat iets juist of onjuist is aangepast worden.

## Views.Login

Nadat fouten in de UserController opgelost waren, werkte de methodes weer naar behoren.

## Views.Register

Nadat fouten in de UserController opgelost waren, werkte de methodes weer naar behoren.

## Views.Shedule

Nadat fouten in de de TaskController, RepeatingTaskController, SubjectController, GradeController, en AppoinmentController opgelost waren, werkte de methodes weer naar behoren.

# Revisies

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Wat | Wie | Versie |
| 12-06-2017  08:45 CEST | Documentatie van de aanpassingen gedocumenteerd. | Patrick van Batenburg | 1.0.0 |
| 14-06-2017  23:23 CEST | Documentatie van de aanpassingen verbeterd. | Patrick van Batenburg | 2.0.0 |