

# Spezifische Entscheidungen

## StateRepository:

Hinzufügen eines StateRepositorys: Dieses „Repository“ hat noch keine Datenbankimplementierung und dient aktuell bloß als Zwischenspeicher für selectedElements. Grund dafür ist, dass man sonst an beide Services die selectedTour und selectedTourLog übergeben müsste (man müsste in einem ViewModel zwei Services initialisieren). Um das zu vermeiden, greifen die Services jeweils auf das StateRepository zu. Zu Beginn wurde überlegt die Daten über ein Event zu senden, jedoch ist dies keine optimale Entscheidung da man keine „großen“ Daten wie Objekte über ein Event senden sollte.

## WindowManager:

Es wurde ein WindowManager hinzugefügt um ein weiteres Fenster zu öffnen wenn ein bestimmter Button wie beispielsweise „Add Tour“ gedrückt wird. Dieser wird einmalig in der ViewFactory definiert (Singleton). Es ist der zentrale Manager über dem neue Fenster geöffnet und auch wieder geschlossen werden können.

## EventManager:

Der Eventmanager wurde aktuell umgeändert, um auch andere Datentypen wie Boolean übertragen zu können. Hier wurde aktuell ein generischer Typ hinzugefügt, das heißt es könnten auch andere Datentypen übertragen werden. Allerdings wird dies in Zukunft noch geändert, da es nicht möglich sein soll, jeden Datentyp zu übertragen (nur Boolean und String...). In Zukunft werden zwei einzelne Maps erstellt, jeweils eine für String und eine für Boolean Events um überprüfen zu können, ob die Datentypen vom Publisher und Subscriber auch übereinstimmen. Das ist aktuell noch etwas problematisch, da der Entwickler unterschiedliche Datentypen bei Publisher und Subscriber angeben könnte und der Compiler keinen Error werfen würde.

## ControlsFormatter:

Es wurde eine ControlsFormatter Klasse hinzugefügt, die die Controls auf bestimmte Art und Weise beeinflusst.

Aktuelle Funktionen:

- Input field nur für float values (keine Buchstaben und Zeichen, nur Punkt und Zahlen erlaubt)
- TableColumnCutOff: Es wurde eine weitere Funktion implementiert um die Textlänge einer Spalte in der Ansicht zu begrenzen um unnötig lange oder hohe Spalten zu verhindern.

## DetailsView:

Gewisse Spalten wie die Tour Description oder bei den TourLogs der Kommentar können einen längeren Text von mehreren 100 Zeichen enthalten, um diesen Text besser darstellen zu können und vollständig zu sehen wurde eine DetailsView erstellt.

## Services:

Es wurden zwei Services jeweils für Tour und TourLog erstellt um die Logik klar zu trennen. Daten die für beide Services relevant sind, beispielsweise selectedTour welche in TourLog auch benötigt wird, da TourLogs nur angezeigt werden können wenn auch eine Tour ausgewählt wurde, sind im StateRepository zu finden.

## Herausforderung:

Beim editieren einer Tour werden sowohl der alte als auch der neue Stand der aktuellen Tour benötigt. Klickt man auf den Button confirm so werden die Änderungen übernommen allerdings wird während eines Events die aktuelle selectedTour auf null gesetzt, da der Table aktualisiert wird und somit keine Tour mehr ausgewählt wird. Danach müssen allerdings noch alle TourLogs zur zugehörigen Tour aktualisiert werden auf den neuen Key (newTour). Da dann allerdings die aktuelle selected Tour null ist, musste die aktuelle Tour neu gesetzt werden im TourService.

## Weitere Infos:

Alle Views welche zusätzliche geöffnet werden (zusätzliche Fenster) sind im Code unter resources und dann dialogs zu finden.