

# Liste offener Punkte [LoP]

---

## Grundstruktur & Organisatorisches

- Sameln von Satzketzen die thematische Tangenten anschneiden. [kontinuierlich]
  - ☐ Bisherige "Methodik" aufteilen (in Methodik und Bearbeitung/Implementierung) [WIP]
  - ☐ Beschreibung der Methodik überarbeiten [WIP]
- 

## Methodik

- Kapitel über verwendete Tools
    - ☐ Flutter Driver [aufteilen & überarbeiten]
    - ☐ Dart DevTools [aufteilen & überarbeiten]
- 

## Bearbeitung

- Für jedes Kriterium jede ZVM abarbeiten (jeweils Code Snippet erstellen und beschreiben)
  - ☒ Definition, Deklaration und Initialisierung eines Globalen Zustands (201124-1815)
    - ☒ Store, Notifier, BLoC, Actions und Events (201125-16:00)
    - ☒ StoreProvider, BlocProvider, ChangeNotifierProvider etc. (201126-1330)
  - ☒ Codeausschnitt der Abfrage zur Darstellung (201130-1130)
  - ☐ Beschreibung der Abfrage zur Darstellung [WIP]
  - ☒ Codeausschnitt der Zustandsänderung (201130-1430)
  - ☐ Beschreibung der Zustandsänderung [WIP]
- Automatisierte Tests (Flutter Driver):
  - CPU-Taktfrequenz vereinheitlichen (CPU-Governor überbrücken, am Beispiel von Filip Hracek)
  - Test-Code beschreiben
- Händische Tests (Dart DevTools):
  - Test-Vorgehen beschreiben

## Ergebnisse

- Daten sammeln
  - Datenauswertung:
    - Gelesene Daten verstehen/ordnen
      - Entsprechendes Kapitel verfassen
    - Daten als Datei exportieren
    - Analyse-Skript erstellen
    - aussagekräftige Graphiken erstellen
      - Entsprechende Software finden
-

## Layout & Diagramme

- ☐ Messergebnisse ordentlich darstellen
  - Finale Überprüfung des Layouts, nachdem alle Texte geschrieben sind
- 

## Abgearbeitete Punkte: (in chronologischer Reihenfolge)

- Die jeweiligen ZVM implementieren (201021-1155)
  - ☒ setState
  - ☒ Provider
  - ☒ BLoC
  - ☒ Redux (201020-1717)
  - ☒ MobX (201021-1155)
- Einleitung/Exposé überarbeiten (201024-1540)
- Stand der Wissenschaft/Technik beleuchten (201025-1912)
- Integrationstests zur Messung der Renderzeiten für jede ZVM implementiert (201028-1630)
- Kapitel über die Applikation, unabhängig von Zustandsverwaltungsmethoden (folgend ZVM), verfassen
  - Welche Eigenschaften des Frameworks werden genutzt?(201102-1740)
  - Wie interagiert die Softwarearchitektur mit den untersuchten Kriterien (201103-1520)
- App anpassen, um beim Zustandswechsel einen kleineren Teils des Widget-Trees auszutauschen. (Optimierung der Performanz!)(201107-1425)
- Kapitel über die Applikation, unabhängig von Zustandsverwaltungsmethoden erweitern und anpassen:
  - Datenmodell beschreiben (201110-1345)
  - Schaubilder erstellen (201111-1220)
  - Verwendete Tools beschreiben [Rohfassung] (201112-1515)
- Diagrammsoftware wählen (201116- 1000)
- In der Providerimplementierung Selector-Widgets statt Consumer-Widgets verwenden (201117-1650)
- Kapitel über verwendete Tools
  - Schaubilder zur Applikation erstellen (201119-1400)
  - ZVM (201123-1845)
    - Datenfluss der einzelnen ZVM veranschaulichen
  - ZVM APIs untersuchen (201124-1000)
- Zu viele Videospiele die mich ablenken (24/7)