

ID	Funktionale Anforderung (aus Sicht eines Component Assemblers)	Priorität (MusCoW)	Aufwand (h, m, g)	Notwendige Pattern	Notwendige (Java-)Techniken	Anmerkungen
1	Verknüpfen von Komponenten und Abhängigkeiten auflösen dazu (zur Startzeit und Deployment)	C-M		Dependency Injection		
2	Komponenten aus der LZU entfernen	M				
3	Komponenten in die LZU hinzufügen (Deployment)	M	sh		Reflection API; Classloading; OO; String Handling;	
4	Übersicht über laufende und deploybare Komponenten inkl. Statusanzeige	M				
5	Bereitstellen von Events	C-W				
6	Konfiguration von laufenden Komponenten	C-W				
7	Verwaltung von Schnittstelle	C-W				
8	Export einer Konfiguration und des internen Status ("Persistierung"); Ausgabe und Überwachung des Lebenszyklus	C		State Pattern		
9	LZU starten	M				
10	LZU herunterfahren (shutdown)	M				
11	Komponenten in der LZU starten	M				
12	Komponenten in der LZU stoppen	M				
13	Parallele Ausführung von unterschiedlichen Komponenten	M			Java Threads	
14	Parallele Ausführung von verschiedenen Instanzen einer Komponente	M			Java Threads	Stop einer Komponente / Threads ("Interrupted")
15	Load-Balancing	W				
16	Bereitstellung von Diensten (z.B. Logging)	C		Dependency Injection		
17	Auf Fehlerhafte Komponenten reagieren / Exception-Handling (z.B. / u.a. Recovery)	C-M				
18	Management-Schnittstelle (CLI; Web-API)	M		Command Pattern		

Legende Prio:

M = Übung Nr. 2

C = Weitere Übung ab Nr. 3

W = Nice to Have, ggf. für Semesterprojekt relevant