

USER ACCEPTANCE TESTING TEST CASE TEMPLATE	
--	--

PROJECT NAME	Energiemanagement-Dashboard	TESTING START TIME	13:00:00
TESTING START DATE	27.06.25	TESTING END TIME	17:45:00
TESTING END DATE	29.06.25		
NAME OF TESTER(S)	Kimia Jamei		

Test No.	Description of Tasks	Steps to Execute	Expected Results	Pass	Fail	Defect / Comments / Additions
Application 1: ChartObserver						
1	Registrierung des ChartObservers	Device anlegen, Observer hinzufügen	Observer wird der Liste hinzugefügt	X		
2	update() verarbeitet neue Daten	Werte zu einem Device hinzufügen → Observer benachrichtigt	Chart-Update durch Observer erfolgt	X		
3	updateAnalysis() berechnet Durchschnitt	Analyse-Dropdown auf "Durchschnitt" setzen und prüfen	Analyseanzeige zeigt richtigen Wert	X		
Application 2: Device						
1	Device-Objekt korrekt erstellen	Neues Device mit Name & Einheit instanzieren	Name & Einheit stimmen	X		
2	Messwert hinzufügen	addMeasurement() mit Zahl aufrufen	Wert wird im Array gespeichert	X		
3	Mehrere Messwerte	Zwei Messwerte hinzufügen, dann getMeasurements() aufrufen	Gibt Array mit beiden Werten	X		
Application 3: DeviceManager						
1	Fügt ein Gerät hinzu und prüft, ob es gefunden wird	1. Gerät erstellen 2. Mit addDevice() hinzufügen 3. Mit getDevice() nach Name suchen	Gerät wird gefunden (undefined ist <b>nicht</b> zurückgegeben)	X		
2	Gibt alle Geräte zurück	1. Zwei Geräte mit addDevice() hinzufügen 2. getAllDevices() aufrufen	Rückgabe enthält <b>genau 2 Geräte</b>	X		
3	Entfernt ein Gerät erfolgreich	1. Gerät mit addDevice() hinzufügen 2. removeDevice() aufrufen	Rückgabe ist true, Gerät existiert nicht mehr	X		
4	Löscht alle Geräte	1. Mehrere Geräte hinzufügen 2. clear() aufrufen 3. getAllDevices() aufrufen	Rückgabe ist ein <b>leeres Array</b>	X		
5	Fehler bei Hinzufügen eines Geräts mit gleichem Namen	1. Gerät mit addDevice() hinzufügen 2. Gleiches Gerät nochmal hinzufügen	Es wird ein Error geworfen mit entsprechender Fehlermeldung	X		
6	Gibt undefined zurück, wenn ein Gerät nicht existiert	getDevice("nichtVorhanden") aufrufen	Rückgabe ist undefined	X		
7	Gibt false zurück, wenn zu entfernendes Gerät nicht existiert	removeDevice("nichtVorhanden") aufrufen	Rückgabe ist false	X		
Application 4: PeakStrategy						
1	Testet, ob PeakStrategy den höchsten Wert aus mehreren Messwerten ermittelt	1. PeakStrategy-Instanz erstellen 2. Messwerte mit Werten [1, 7, 5] definieren 3. analyze() aufrufen	Rückgabewert ist 7	X		
2	Testet, ob bei leerer Liste 0 zurückgegeben wird	1. PeakStrategy-Instanz erstellen 2. Leere Liste übergeben 3. analyze() aufrufen	Rückgabewert ist 0	X		
Application 5: SavingPotentialStrategy						
1	Testet Sparpotenzial-Berechnung mit [100, 200, 300]	1. Neue Instanz erzeugen 2. analyze([100, 200, 300]) aufrufen	Rückgabe: "Sparpotenzial: ~40.0 W"	X		
2	Testet Verhalten bei leerem Messwerte-Array	1. Neue Instanz erzeugen 2. analyze([]) aufrufen	Rückgabe: "Keine Daten"	X		
3	Testet Rundung bei nicht-ganzzahligem Ergebnis	1. Neue Instanz erzeugen 2. analyze([10, 15]) aufrufen	Rückgabe: "Sparpotenzial: ~2.5 W"	X		
4						
5						
Application 6: Subject						
1	Observer-Verwaltung (add, remove, notify)	1. Observer anhängen 2. notify("Test") aufrufen	Observer erhält Nachricht "Test"	X		
2	Observer-Verwaltung (add, remove, notify)	1. Observer anhängen 2. entfernen 3. notify("Test") aufrufen	Observer wird <b>nicht</b> benachrichtigt	X		

GENERAL QUESTIONS / COMMENTS	