Kontrakte

**[Administration]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | addBooking(chargingZone, user, maxCapacity, plugType, startTime, endTime, socStart)  Erstellen und hinzufügen einer Buchung  Administration besteht  Buchung wurde erstellt und der Administration hinzugefügt |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert, ob die Buchung akzeptiert wurde.  Ungültige Ladezone, Nutzer, MaxKapazität, Steckertyp, Startzeit, Endzeit  -  Systemoperation  -  Domi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | removeBooking(id)  Entfernen einer Buchung  Mindestens eine Buchung in Administration vorhanden; Id muss valide sein  Buchung wurde aus der Administration entfernt. |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Buchung entfernt wurde  Angeforderte Buchung nicht vorhanden  -  Systemoperation  -  Domi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | editBooking(id, chargingZone, maxCapacity, plugType, startTime, endTime)  Bearbeiten einer Buchung  Mindestens eine Buchung in Adminstration vorhanden  Attribute der Buchung wurde verändert |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Buchung erfolgreich geändert wurde  Angeforderte Buchung nicht vorhanden  -  Systemoperation  -  Domi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | makeSuggestion(booking)  Erstellt eine Vorschlaf für eine Buchung  Eine Buchungsanfrage wurde abgelehnt;  Buchung wurde erstellt und er der Nutzer benachrichtigt Anhand der Buchung b wurde der best mögliche Ladeplatz-Vorschlag erstellt |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Ladeplatz-Vorschlag für die Buchung  -  -  Systemoperation  -  Uli |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | updateTimetable(bookings)  Fügt die Buchungen bs in die Timetable hinzu und aktualisiert diese  Buchungen bs wurden erfolgreich angelegt und bestätigt  Buchungen bs wurden erstellt Buchungen bs wurden der Timetable hinzugefügt |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Neues Set an Buchungen bs  -  **-**  Systemoperation  -  Uli |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | loadBookings()  Lädt alle Buchungen erneut ein  -  Buchungen wurden geladen |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Buchungen erfolgreich geladen wurden  -  Gibt die Werte der Buchung an die View zurück  Systemoperation  Buchungsübersicht anzeigen  Uli |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | saveBooking()  Speichert die Änderungen an einer Buchung  Buchung wurde erstellt  Änderungen an der Buchung (falls vorhanden) wurden gespiechert Timetable wurde aktualisiert |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Speichert die Daten zu einer Booking ab  -  -  Systemoperation  Buchung anlegen/bearbeiten  Uli |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | messageUser(user)  benachrichtig den User unter mehreren Umständen  user muss existieren  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | -  -  Schickt Nachricht an User  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | checkIn(id)  Nutzer bestätigt, dass er seine Buchung wahrnimmt; Überprüft auch ob er Zeit einhält  Buchung mit der entsprechenden ID muss existieren  StartTime liegt max. 15 Min in der Zukunft  endTime lieg in der Zukunft |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Checkin erfolgreich war  Id existiert nicht  User wurde Erfolgreich eingecheckt  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | checkOut(id)  Nutzer bestätigt, dass er den Ladestandtort verlässt  Buchung mit der entsprechenden ID muss existieren; muss sich vorher einchecken  Buchung liegt in der Vergangenheit |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Checkout erfolgreich war  Id existiert nicht  User wurde Erfolgreich ausgecheckt  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | getPriority(email)  es wird die Priorität der Buchungsanfrage des Nutzers bestimmt  email muss existeren  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Priorität  -  -  Systemoperation  -  Thomas |

**[Timetable]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | scheduleBookings()  Verteilt die aufgegebenen Buchungsanfragen und bestätigt diese oder lehnt diese ab  Es muss mindestens eine Buchungsanfrage existieren  Verteilung der Buchungen hat stattgefunden |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Verteilung erfolgreich war  -  Buchungsreihenfolge kann nun eingesehen werden  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | scheduleAdhocBookings()  Verteilt die aufgegebenen Buchungsanfragen sofort auf die noch freien Slots und bestätigt diese oder lehnt diese ab  Es muss mindestens eine Buchungsanfrage existieren  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Verteilung erfolgreich war  -  Buchungsreihenfolge kann nun eingesehen werden und wurde aktualisiert  Systemoperation  -  Thomas |

**[Infrastructure]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | addLocation(name)  -  -  Location existiert |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | eine Localisation wurde kreiert und er Infrastruktur hinzugefügt  Es wurde kein name übergeben.  -  Systemoperation  -  Peter |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | removeLocation(id)  Löscht eine Location anhand der ID  Id muss existieren; Eine Location muss in der Liste existieren  Die Location inklusive ihrer Id wird gelöscht, sowie alle dazugehörigen Charging Zones, Stations und plugs. Zusätzlich wird die Location aus jeder Liste in den Infrastrukturen entfernt |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob Location erfolgreich gelöscht wurde  Es wurde keine Location mit passender ID gefunden  -  Systemoperation  -  Peter |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | Infrastructure()  Konstruktor  Ruft getInstance() auf  -  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | -  -  -  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | getInstance()  Erstellt ein neues Infrastructure Objekt  Es darf kein Infrastructure Objekt existieren  Es existiert jetzt ein Infrastructure Objekt |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Es existiert ein Infrastructure Objekt  -  -  Systemoperation  -  Thomas |

**[Location]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | addChargingZone(site, maxPowerZone)  Eine neue ChargingZone wird kreiert mit einer MaxPowerZone;  Es müssen bereits alle Buchstaben des Alphabets bis hin zu diesem in der site bereits einer site in der Localisation zugeordnet sein  eine site wird zugeordnet, eine MaxpowerZone wird erstellt; Die MaxPowerzone muss in einem gewissen Intervall bleiben |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | eine ChargingZone mit Attributen wird erstellt  -  -  Systemoperation  -  Peter |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | removeChargingZone(id)  entfernt die Ladezone der übergebene Id in Locations und deren Ladestationen  es existiert mindestens eine Ladezone und deswegen eine Infrastruktur  Ladezone mit id wurde entfernt und daher alle Ladestationen in der Ladezone |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob entfernen erfolgreich ist  Keine Ladezone der übergebenen id wurde gefunden  -  Systemoperation  Editing ein existierendes Szenario  Thao |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | editCharginZone(id, site)  Ladezone mit id wird geändert  Es existiert mindestens eine Ladezone und daher eine Location und eine Infrastruktur  Die Ladezone mit der der übergebenden Id wurde geändert |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob ändern erfolgreich war  Keine Ladezone mit der übergebenen Id wurde gefunden  -  Systemoperation  Editing ein existierendes Szenario  Thao |

**[ChargingZone]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | addChargingStation(manufacturerg, name, maxPower)  fügt eine neue Ladestation mit Herstellername, Name, maxPower und verfügbare Stecker hinzu und fügt diese der Liste hinzu  es existiert mindestens eine Ladezone und daher eine Location und eine Infrastruktur  Ladezonenliste von Ladestationen haben einen neuen Eintrag  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob hinzufügen erfolgreich ist  -  -  Systemoperation  Editing eine existierendes Szenario  Thao |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | removeCharingStation(id)  entfernt eine Ladestation mit der übergebenen id von einer Ladezone  es existiert mindestens eine Ladezone und deswegen eine Location und Infrastruktur  die Ladestation mit der übergebenen id wurde von der Ladezone entfernt und die maximale Power der Ladezonen wurde rekalkuliert |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | -  Keine Ladestation mit der übergebene id wurde gefunden  - Systemoperation  Editing ein existierendes Szenario  Thao |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | editCharginStation(manufacturer, name, maxPower)  verändert eine existierende Ladestation der übergebenen Id  es existert mindesetens eine Ladezone und deswegen eine Location und Infrastruktur  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Wahrheitswert ob ändern erfolgreich war  Keine Ladestation mit der übergebenen Id wurde gefunden  -  Systemoperation  Editing ein existierendes Szenario  Thao |

**[ChargingStation]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | addPlug(type, power)  Fügt einen neuen Stecker zu einer Ladesäule hinzu  Der Typ muss aus der Liste der gültigen Typen sein.  Power muss größer 0 sein.  Stecker wurde erfolgreich angelegt und ist von nun an unter einer bestimmten Id abrufbar |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Gibt zurück ob anlegen erfolgreich war  Typ ist nicht korrekt  -  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | removePlug(id)  Entfernt einen bestimmten Stecker von einer Ladesäule anhand der ID  Id muss gültig sein  Stecker wurde erfolgreich von der Ladesäule entfernt |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Gibt Wahrheitswert zurück ob entfernen erfolgreich war  Stecker mit der übergebenen Id wurde nicht gefunden  -  Systemoperation  -  Thomas |

**[Simulation]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | addSecenario(duration, rushhours, spread, minRequests, maxRequests)  Erstellen von einem neuen Szenario in bestimmter Länge, mit der Angabe keiner, einer oder mehrerer Rushhour/s, einer Verteilung, minimaler und maximaler Anfragen  -  Szenario wurde erfolgreich mit allen erforderlichen Attributen angelegt und ist von nun an unter einer Bestimmten Id abrufbar |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Gibt Wahrheitswert zurück ob Szenario erfolgreich angelegt wurde  -  Duration muss größer 0 sein;  Rushhours, kann keine oder mehrere Zeitangaben beinhalten;  Spread darf muss zwischen 0 und 1 liegen;  minRequests muss kleiner gleich maxRequests sein und nicht negativ;  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | editScenario(id, completed, duration, rushours, spread, minRequests, maxRequests)  Ändern eines bestimmten Szenarios mit passenden Id;  Die Länge des Szenarios, Rushhour/s, einer Verteilung, minimaler und maximaler Anfragen kann geändert werden  Szenario mit entsprechender Id muss existieren  Completed wird auf false gesetzt Szenario wurde erfolgreich mit allen erforderlichen Attributen geändert und ist von nun an unter der gleichen Id abrufbar |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Gibt Wahrheitswert zurück ob Szenario erfolgreich geändert wurde  Szenario mit der entsprechenden Id wurde nicht gefunden  Duration muss größer Null sein;  Rushhours, kann Null oder mehrere Zeitangaben beinhalten;  Spread muss zwischen 0 und 1 liegen;  minRequests muss kleiner gleich maxRequests sein und beides nicht negativ;  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | runScenario(id)  Es wir das entsprechende Szenario ausgeführt  Es muss eine Location erstellt worden sein und mindestens ein Szenario muss der Simulation hinzugefügt worden sein.  für jede Ladezone und Ladestation ist eine Auslastung pro tick Vorhanden, sowie eine Gesamtauslastung. |
| **Ergebnisse:**  **Ausnahme:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkung:** | Auslastung der Ladezonen und Ladestationen pro tick.  Keine Location/unvollständig  Kein Szenario  -  Systemoperation  -  Tobi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | saveSzenarios()  Persistiert die Szenarien aus der Simulation  Es müssen Szenarien in der Historie enthalten sein.  Die Szenarien sind in einer JSON-Datei persistiert |
| **Ergebnisse:**  **Ausnahme:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkung:** | -  die Historie enthält keine Szenarien  -  Systemoperation (Speicheroperation)  -  Tobi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | loadSzenarios(JSON-Datei)  Liest eine JSON-Datei ein und lädt die Enthaltenen  Szenarien. Die Szenarien werden der Historie hinzugefügt.  Es müssen schon mal Szenarien gespeichert worden sein.  Die Liste der Szenarien, die in der Historie enthalten sind, wurde um die Szenarien aus der JSON-Datei erweitert. |
| **Ergebnisse:**  **Ausnahme:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkung:** | -  Es gibt einen Fehler beim Einlesen der Szenarien  -  Systemoperation (Speicheroperation)  -  Tobi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | Simulation()  Konstruktor  Ruft getInstance() auf  -  - |
| **Ergebnisse:**  **Ausnahme:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkung:** | -  -  -  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | getInstance()  Erstellt ein neues Simulation Objekt  Es darf kein Simulation Objekt existieren  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Es wurde ein Simulation Objekt kreiert  -  -  Systemoperation  -  Thomas |

**[History]**

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | evaluateSzenario(id)  Die Auslastung der Ladezonen und Ladestationen die bei runSimulation() wird erstellt wird, wird ausgewertet und graphisch dargestellt  runSimulation() wurde über das Szenario ausgeführt.  - |
| **Ergebnisse:**  **Ausnahme:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | -  Übergebenes Szenario wurde evaluiert  die UI erhält die ausgewerteten Daten um sie zu veranschaulichen  Systemoperation  -  Tobi |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | Histroy()  Konstruktor  Ruft getInstance() auf  -  - |
| **Ergebnisse:**  **Ausnahme:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | -  -  -  Systemoperation  -  Thomas |

|  |  |
| --- | --- |
| **Operation:**  **Beschreibung:**  **Vorbedingung:**  **Nachbedingung:** | getInstance()  Erstellt ein neues History Objekt  Es darf kein History Objekt existieren  - |
| **Ergebnisse:**  **Ausnahmen:**  **Ausgaben:**  **Typ:**  **Querverweise:**  **Anmerkungen:** | Es wurde ein History Objekt kreiert  -  -  Systemoperation  -  Thomas |