

Server seite:

Der Server kann mit 3 verschiedenen Clienttypen verbunden sein:

1. Client: ist dafür verantwortlich, die Personen in Serverdatenbank zu verändern.
2. Pi Client: im Fall eines Einsatzes, zeigt er die Ankunftszeitinformationen
3. Android Client: kann die ankunftszeit für die betroffene Person ändern

Die Kommunikation geschieht durch Tauschen von JSON Objects, Befehle kann man steuern durch JSON property „action“, Befehle können auch weitere Informationen brauchen (Parameter), in diesem Fall müssen die Parameter auch in dem gleichen JSON object rein gepackt werden.

Befehle auf der Serverseite:

Action	Parameter	Beschreibung	Weiterleiten an
auth	token : String (security token)	Erstellt eine Verbindung zwischen Server und Client. Befehle können ohne auth gesendet werden, müssen aber einen richtigen Token haben, um Nachrichten vom Server zu bekommen. Braucht man ein auth.	
	type : String ("android", "client", or "pi")		
add person	person : Person object	Fügt die Person in die Datenbank hinzu	Clients
	token : String (security token)		
delete person	person : Person object	Löscht die Person aus der Datenbank, falls vorhanden	Clients
	token : String (security token)		
update person	person : Person Object	Ändert die Daten einer Person in der Datenbank, falls die Person nicht vorhanden ist, wird sie hinzugefügt	Clients
	token : String (security token)		
start mission	mission : Mission object	Beginnt einen Einsatz mit allen Personen auf dem Server	Androids, Clients, Pis
	token : String (security token)		
end mission	token : security token	Endet den Einsatz, falls vorhanden	Androids, Clients, Pis
update arrival time	person : Person object	Ändert die erwartete Ankunftszeit einer Person	Pis
	arrival time: Estimated_time		
	token : String (security token)		

