

# Trip Planner Persistence

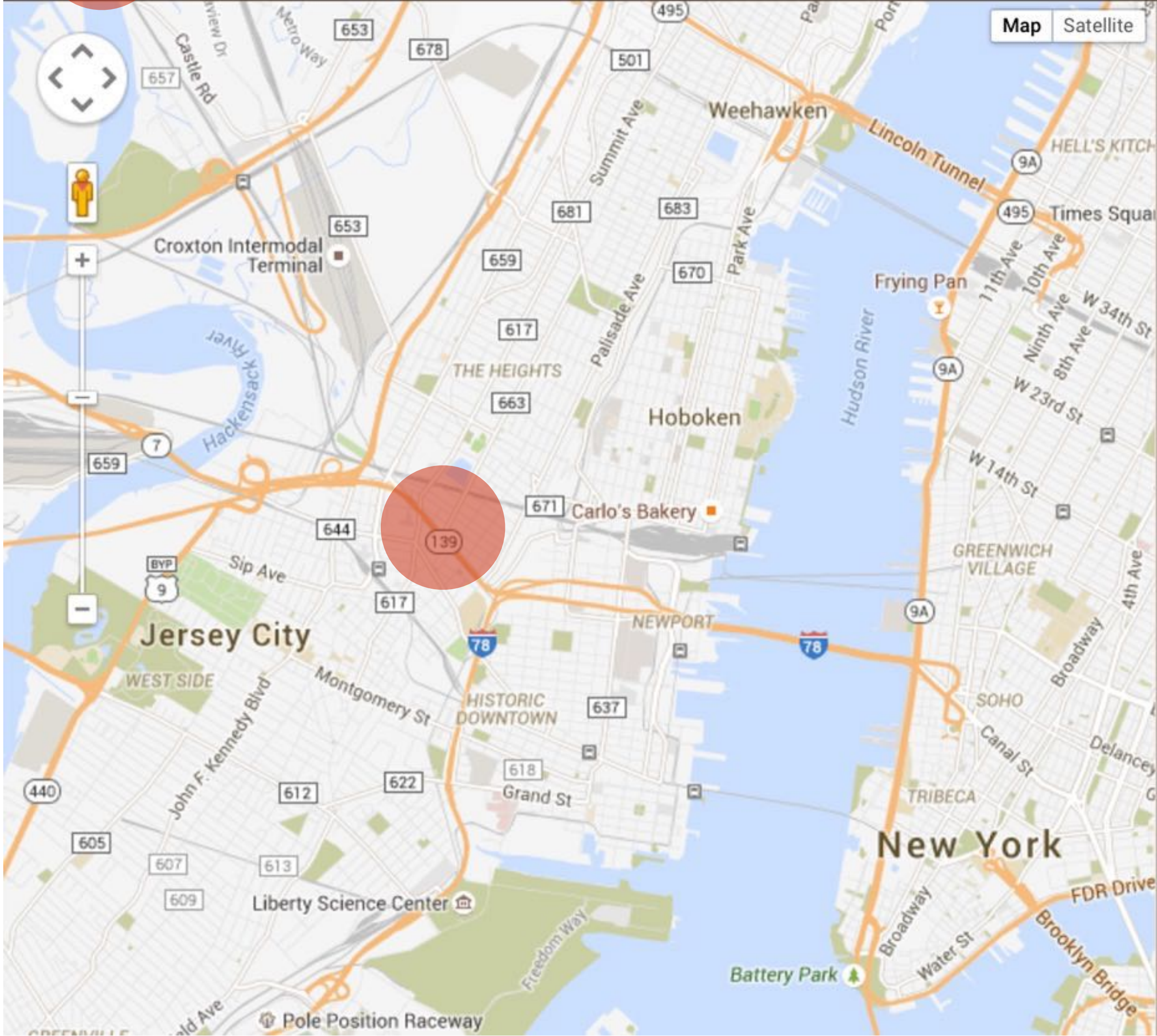
---

*RESTful Patterns*

Method	URI	/api/trip/1	/api/trip/1/days/1	/api/trip/1/days/1/restaurants/1
GET	/	get all trips	get all days for trip 1	get all restaurants for day 1 of trip 1
GET	/id	get trip 1	get day 1 for trip 1	get restaurant 1 for day 1 of trip 1
POST	/	make a new trip	make a new day for trip 1	add a restaurant to day 1 of trip 1
PUT	/id	update trip 1	update day 1 for trip 1	update restaurant 1 for day 1 of trip 1
DELETE	/id	delete trip 1	delete day 1 of trip 1	remove restaurant 1 from day 1 of trip 1
DELETE	/	delete all trips	delete all days in trip 1	remove all restaurants for day 1 of trip 1

# UI?





Hotels

Andaz Wall Street

Restaurants

Marc Forgione

Activities

South Street Seaport

Day 1

1

My Hotel

My Restaurants

My Activities



**For the sake of this review...**

Method	URI	/api/trip/1	/api/trip/1/days/1	/api/trip/1/days/1/restaurants/1
GET	/	get all trips	get all days for trip 1	get all restaurants for day 1 of trip 1
GET	/id	get trip 1	get day 1 for trip 1	get restaurant 1 for day 1 of trip 1
POST	/	make a new trip	make a new day for trip 1	add a restaurant to day 1 of trip 1
PUT	/id	update trip 1	update day 1 for trip 1	update restaurant 1 for day 1 of trip 1
DELETE	/id	delete trip 1	delete day 1 of trip 1	remove restaurant 1 from day 1 of trip 1
DELETE	/	delete all trips	delete all days in trip 1	delete all restaurants for day 1 of trip 1

Method	URI	/api/trip/ <b>1</b>	/api/trip/1/days/ <b>1</b>	/api/trip/1/days/1/restaurants/ <b>1</b>
GET	/	<b>get all trips</b> → trips	<b>get all days for trip 1</b> → days	get all restaurants for day 1 of trip 1
GET	/id	<b>get trip 1</b> → trip	<b>get day 1 for trip 1</b> → <i>day (populated)</i>	get restaurant 1 for day 1 of trip 1
POST	/	<b>make a new trip</b> → trip	<b>make a new day for trip 1</b> → day? trip? 201?	<b>add a restaurant to day 1 of trip 1</b> → restaurant? day? 201?
PUT	/id	update trip 1	update day 1 for trip 1	update restaurant 1 for day 1 of trip 1
DELETE	/id	delete trip 1	<b>delete day 1 of trip 1</b> → day? trip? 204?	<b>remove restaurant 1 from day 1 of trip 1</b> → restaurant? day? 204?
DELETE	/	delete all trips	delete all days in trip 1	delete all restaurants for day 1 of trip 1

Method	URI	/api/trip/1	/api/trip/1/days/1	/api/trip/1/days/1/restaurants/1
GET	/	<b>get all trips</b> → trips	<b>get all days for trip 1</b> → days	get all restaurants for day 1 of trip 1
GET	/id	<b>get trip 1</b> → trip	<b>get day 1 for trip 1</b> → <i>day (populated)</i>	get restaurant 1 for day 1 of trip 1
POST	/	<b>make a new trip</b> → trip	<b>make a new day for trip 1</b> → <i>trip</i>	<b>add a restaurant to day 1 of trip 1</b> → <i>day (populated)</i>
PUT	/id	update trip 1	update day 1 for trip 1	update restaurant 1 for day 1 of trip 1
DELETE	/id	delete trip 1	<b>delete day 1 of trip 1</b> → <i>trip</i>	<b>remove restaurant 1 from day 1 of trip 1</b> → <i>day (populated)</i>
DELETE	/	delete all trips	delete all days in trip 1	delete all restaurants for day 1 of trip 1



# NOTE!

- While the basic REST patterns are relatively straightforward, the precise behavior / which routes to use is more nuanced
- We have made ad-hoc decisions to simplify our front-end code
- What endpoints should return is debatable
- Where and when to populate
- Currently ignoring status codes (e.g. created ought to be 201)
- Is adding sub-resource the same as updating parent resource?
- Use this demo / solution as a guide, not “the rules”
- <http://www.vinaysahni.com/best-practices-for-a-pragmatic-restful-api>