Apis:

Online domain:
https://back-end-new-api.azurewebsites.net/

Path	Type	Request Payload	Response	Explanation

/rt_parking_info	POST	{
		“parking_lot_id”: xxx.
		,
		“owner”:xxxxx
		
		//for test purpose i am Gonna have couple test id and owner for each of the corresponding stream in the video
		
		https://docs.google.com/document/d/19YYrMI6yazP764tIJ50J-8urihMEMhSdKpmKCEqM50E/edit
		, so just pass that, and model will be looking at those specific parking lot
		}
			{
			"parking_lots":
			{
			"id": 1,
			"owner": "5dert6",
			"parking_spaces": [
			{
			"id": 0,
			"status": true,
			"camcoords":
			[[[551,431],[555,513],[683,516],[654,433]]
			],
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 1,
			"status": true,
			"camcoords":
			[[444,437],[429,521],[548,526],[545,440]]
			,
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 2,
			"status": false,
			"camcoords":
			[[342,442],[311,530],[414,537],[439,454]]
				For real time info (later for final product)
				
				0: standard
				1: handicapped
				2: reserved

			,
			"mapcoords": [],
			"type": 1
			},
			{
			"id": 3,
			"status": true,
			"camcoords":
			[[228,454],[197,540],[300,543],[317,472]]
			,
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 4,
			"status": true,
			"camcoords":[[228,454],[197,540],[300,543],[317,472]]
			,
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 5,
			"status": true,
			"camcoords":[[228,454],[197,540],[300,543],[317,472]]
			,
			"mapcoords": [],
			"type": 0
			}
			]
			}	

			}	

localhost:port/sp_parking_info	GET	{
		“parking_lot_id”: xxx.
		,
		“timestamp”:xxxxx’
		,
		‘photo_id’: xxx
		}
			{
			"parking_lots":
			{
			"id": 1,
			"owner": "5dert6",
			"parking_spaces": [
			{
			"id": 1,
			"status": true,
			"camcoords": [],
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 2,
			"status": true,
			"camcoords": [],
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 3,
			"status": false,
			"camcoords": [],
			"mapcoords": [],
			"type": 1
			}
			]
			}
				Use photo as input sources
				(POC purpose)

/get_prev_layout	GET	{
		“id”:0,
		“name”: owner_name
		}
			{
			"parking_lots":
			{
			"id": 1,
			"owner": "5dert6",
			"parking_spaces": [
			{
			"id": 1,
			"status": true,
				This is for when admin goes back to editing page and it shows what they have done in the past

			"camcoords": [],
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 2,
			"status": true,
			"camcoords": [],
			"mapcoords": [],
			"type": 0
			},
			{
			"id": 3,
			"status": false,
			"camcoords": [],
			"mapcoords": [],
			"type": 1
			}
			]
			}	

/save_coord	POST	{ "parking_lots": { "id": 1, "owner": "5dert6", "parking_spaces": [{ "id": 1, "status": true, "camcoords": [], "mapcoords": [], "type": 0 }, { "id": 2, "status": true, "camcoords": [], "mapcoords": [], "type": 0 },		This is when two view completed their drawing boxes and submit the data to backend to save to the database

		{
		"id": 3,
		"status": false,
		"camcoords": [],
		"mapcoords": [],
		"type": 1
		}
		]
		}		

~~/get_all_parking_lot_id~~ ~~*404~~	~~GET~~	~~{~~ ~~“name”: owner_name~~ ~~}~~	~~{~~ ~~Parking_lot_id = [1,2,3,4,5]~~ ~~}~~	~~This returns all the parking lot id under certain owner, front end only needs to look at the last index value and increment on that to create new parking_lot_id~~

"/get_parking_snapshot"	GET	path&name=???
		{
		url: (from previous creating step)
		}		

/get_analytics	GET	{
		“Id”:0,
		“name”: owner_name
		}
			https://github.com/parkd-app/park-d/blob/analytics_v3/src/park-d-front-end/src/data/analytics.json	

/create_parking_lot	Post	{
		“Id”:0,
		“name”: owner_name
		“url” :
			{
			“result_true”
			}	

		www.youtube.com }		

/get_all_parking_lots	GET	{
		}
			{
			“Parking_lots”: [
			{“name”: odert, “id”: 5},
			{“name”: odert, “id”: 6}
			]
			}	