Requirements

Please provide production quality code and details on deployment strategy.

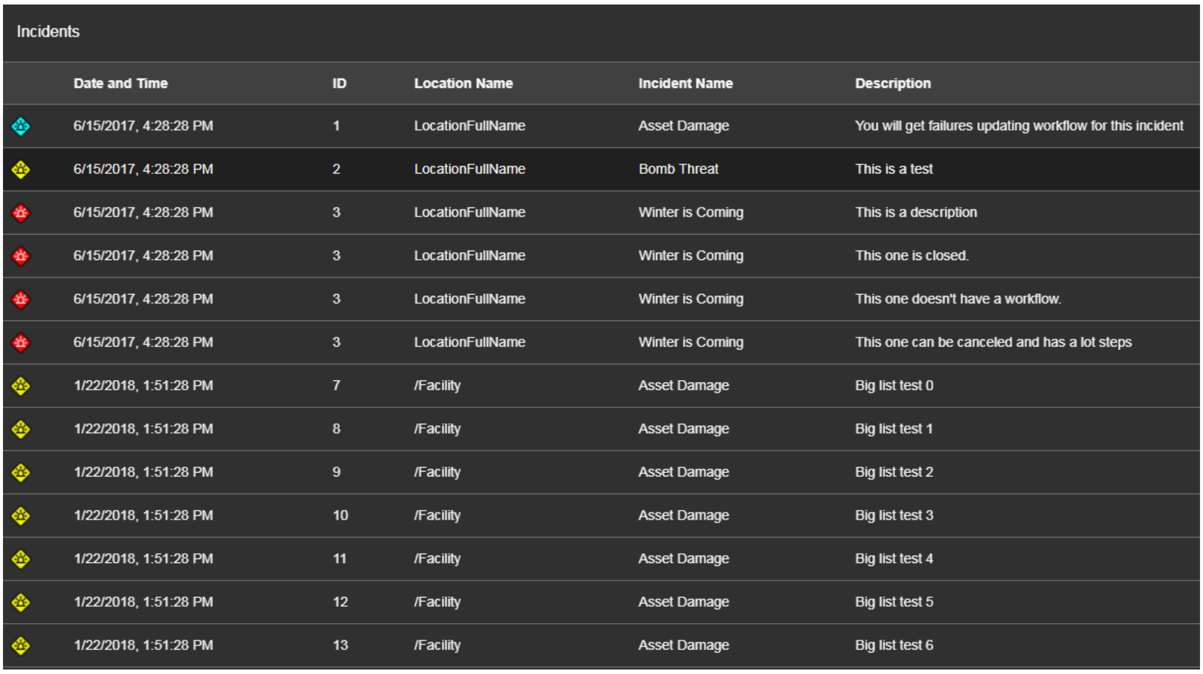
A fake API is provided in src/js/. It contains two methods: getLocations and getIncidentsByLocationId

- Get all incidents from the fake API, order by Priority ascending and Date Time descending. Id is the unique identifier of an incident, duplicate incidents must be filtered out
- Display a table with following columns
    - Icon represents priority, see src/img
    - Incident name, e.g Fire
    - Date time in local format, e.g. 1/22/2018, 10:25:18 PM
    - Priority, e.g. High. Priority values are mapped to the following strings: 1: High, 2: Medium, 3: Low
    - Location name, e.g. T1 Lobby

- If the width of the browser window is less than 600px, display a list instead
- Background colour of hovered rows is indicated in the screenshot
- You must not modify the content of provided files, but you can restructure the folders if you wish
- The application should work in latest Chrome
- Using 3rd party libraries/frameworks is allowed. Note our current technology stack predominantly includes React.js and Knockout.js and we expect growth in React Native for Windows use in future.
- Please create a solution that reflects production quality and is suitable for deploying to a customer
- Share us all your source files so we can compile, build and give the program a go
- Let us know how much time you spent on this challenge
- The goal of the exercise is to get a feel for how you would go about designing and building a high quality component and then be able to discuss it and listen/respond to any feedback. We will review and ask questions about design/coding choices and ask questions about potential improvements. This will be more of an open-ended conversation that simulates how we do code reviews as part of our standard development practices

Screenshots

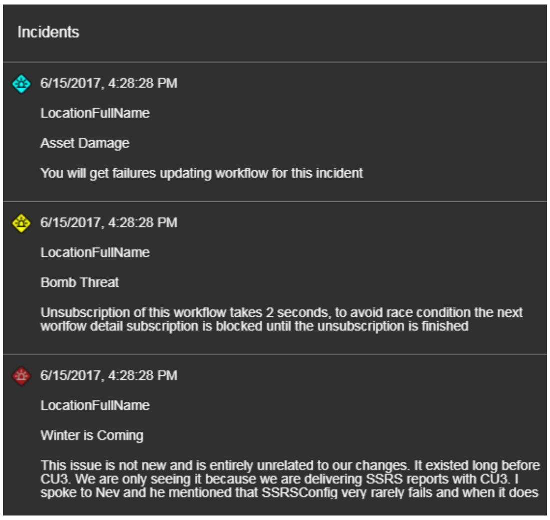
Table



Incidents

	Date and Time	ID	Location Name	Incident Name	Description
◆	6/15/2017, 4:28:28 PM	1	LocationFullName	Asset Damage	You will get failures updating workflow for this incident
◆	6/15/2017, 4:28:28 PM	2	LocationFullName	Bomb Threat	This is a test
◆	6/15/2017, 4:28:28 PM	3	LocationFullName	Winter is Coming	This is a description
◆	6/15/2017, 4:28:28 PM	3	LocationFullName	Winter is Coming	This one is closed.
◆	6/15/2017, 4:28:28 PM	3	LocationFullName	Winter is Coming	This one doesn't have a workflow.
◆	6/15/2017, 4:28:28 PM	3	LocationFullName	Winter is Coming	This one can be canceled and has a lot steps
◆	1/22/2018, 1:51:28 PM	7	/Facility	Asset Damage	Big list test 0
◆	1/22/2018, 1:51:28 PM	8	/Facility	Asset Damage	Big list test 1
◆	1/22/2018, 1:51:28 PM	9	/Facility	Asset Damage	Big list test 2
◆	1/22/2018, 1:51:28 PM	10	/Facility	Asset Damage	Big list test 3
◆	1/22/2018, 1:51:28 PM	11	/Facility	Asset Damage	Big list test 4
◆	1/22/2018, 1:51:28 PM	12	/Facility	Asset Damage	Big list test 5
◆	1/22/2018, 1:51:28 PM	13	/Facility	Asset Damage	Big list test 6

List



Incidents

◆ 6/15/2017, 4:28:28 PM

LocationFullName

Asset Damage

You will get failures updating workflow for this incident

◆ 6/15/2017, 4:28:28 PM

LocationFullName

Bomb Threat

Unsubscription of this workflow takes 2 seconds, to avoid race condition the next worlfow detail subscription is blocked until the unsubscription is finished

◆ 6/15/2017, 4:28:28 PM

LocationFullName

Winter is Coming

This issue is not new and is entirely unrelated to our changes. It existed long before CU3. We are only seeing it because we are delivering SSRS reports with CU3. I spoke to Nev and he mentioned that SSRSConfig very rarely fails and when it does