UWP Code Challenge

We'd like to have you implement a common use case. Please use UWP (C# & XAML) technology to implement the solution targeting any recent Windows OS version.

Please don't use any 3rd party components, controls, or frameworks. Restrict your solution to use only what's available in Visual Studio via File, New Project. One allowance for 3rd party tools is JSON deserialization; feel free to use any tool you are willing to recommend as an ideal technique.

Although the assignment is fairly limited in scope, treat it as a feature in a larger, real-world application, where client-side architecture is important. Quality is more important than quantity, so good architectural and coding decisions are more important than completing any of the extra credit tasks. Once the assignment is complete, we'll review your work to assess the technical decisions made, and the completeness and quality of the work.

Note, we are detail oriented. Please self-review your solution before submitting it to us for review.

Steps

1. Review the requirements below and ask any questions you have via email.
2. Send an email to your technical contact with an estimate of time to completion.
    - Please estimate both total hours, and the projected completion date.
    - Our intent is not to take too much of your time, so try to scope a solution that can be completed within a reasonable effort for this interview purpose.
    - In the subsequent code review, we'll talk about any change/extension you recommend if this was a real project with more time.
3. Complete the coding challenge outlined below; feel free to ask any additional questions via email during this phase.
4. Send the code to us for review.
    - Hopefully this is at (or near) the estimated completion date you provided.
    - You can send a zip file via email, or a share a URL to us to download.

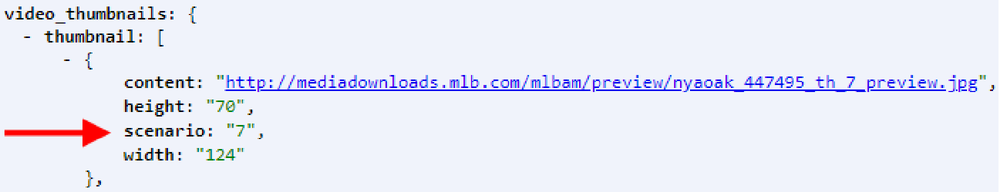
Requirements

Parse out the data for each game in the JSON feed below, and construct a simple list of images that the user can navigate through with the keyboard (or remote), left and right. Assume that a mouse will not be used for input.

API:
http://gdx.mlb.com/components/game/mlb/year_2016/month_05/day_20/master_scoreboard.json

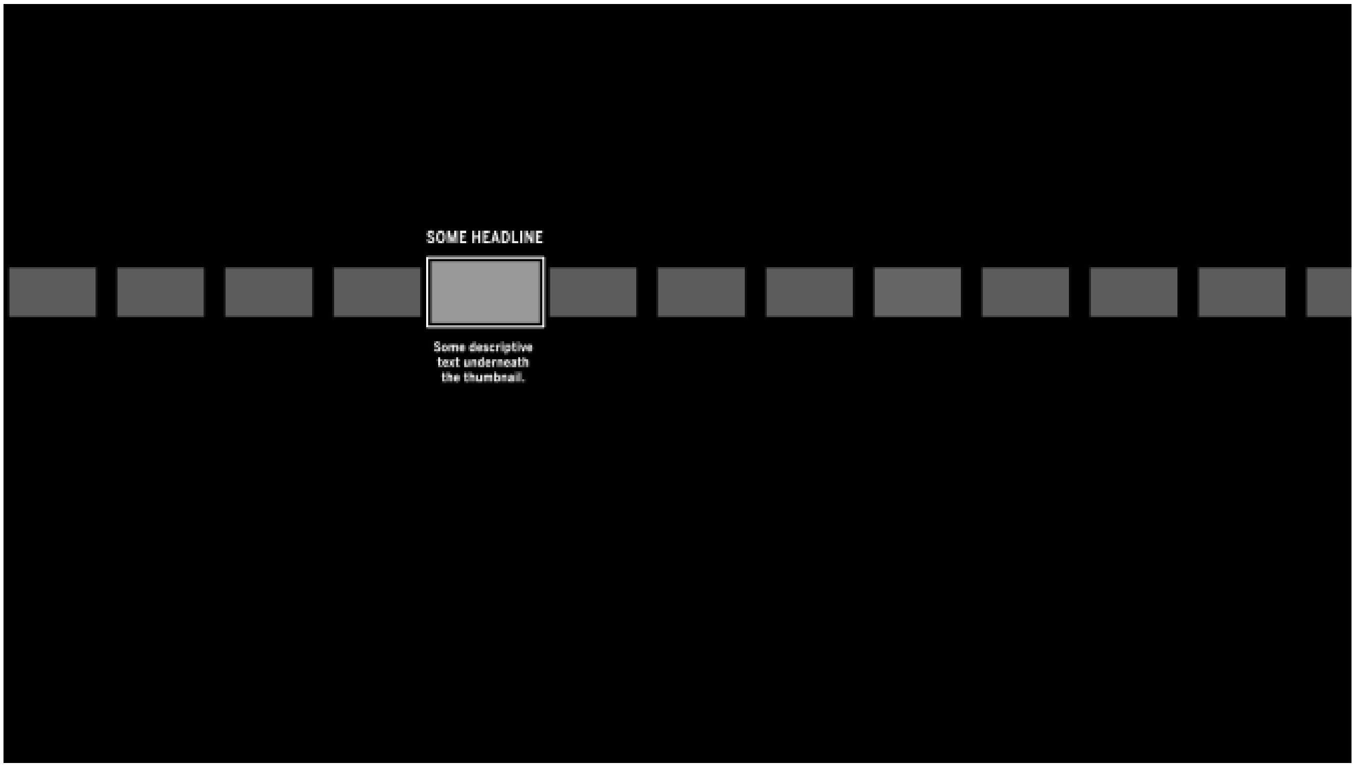
When a game item is focused, the element should scale up by 150% as shown in the wireframe below, and it should display some descriptive metadata of your choosing both above and below the image.

The image displayed in each list item should be the thumbnail having scenario: “7” as shown below:



```
video_thumbnails: {
  - thumbnail: [
      - {
            content: "http://mediadownloads.mlb.com/mlbam/preview/nyaoak_447495_th_7_preview.jpg",
            height: "70",
            scenario: "7",
            width: "124"
        },
```

View Wireframe:



Use this as a background image displayed in the view, behind the list:

http://mlb.mlb.com/mlb/images/devices/ballpark/1920x1080/1.jpg

Extra Credit

1. Allow the user to move to another day
2. Incorporate transitions, animations, or any other visual aesthetics
3. Display a detail view, or overlay, when selecting an item with the keyboard or remote