**UI Designer / Front End Developer Project**

**Problem:**

The Master Physiotherapist would like to review allocated videos offline however, due to compliance issues and patient sensitive data, downloading the videos to their laptop is not an acceptable solution.

**Solution:**

ChilliPharm would like to offer Master Physiotherapists an offline-viewing mode. MPs would select the videos that they would like to view offline. Whilst offline, they would be able to view the videos, mark as Reviewed and complete an online QC Report. These changes would then sync with the system once back online.

Note: The state of a video being reviewed is just a boolean true or false and a QC report can just be a set of input fields with labels ‘xyz’

**The Application:**

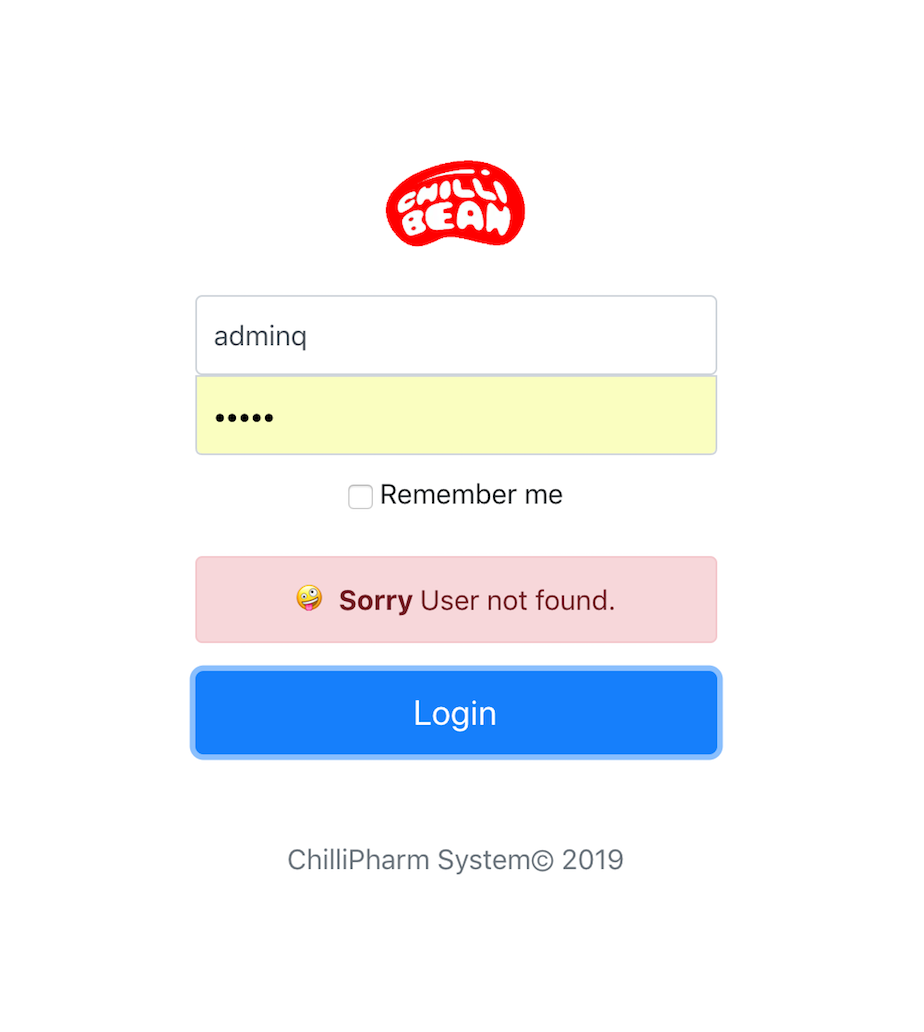
**Login Screen**

Once the application has started the user will be greeted with the Login screen,

The assumption was made the videos will contain sensitive data and that it should be encrypted and secure.

Username & password is **admin**

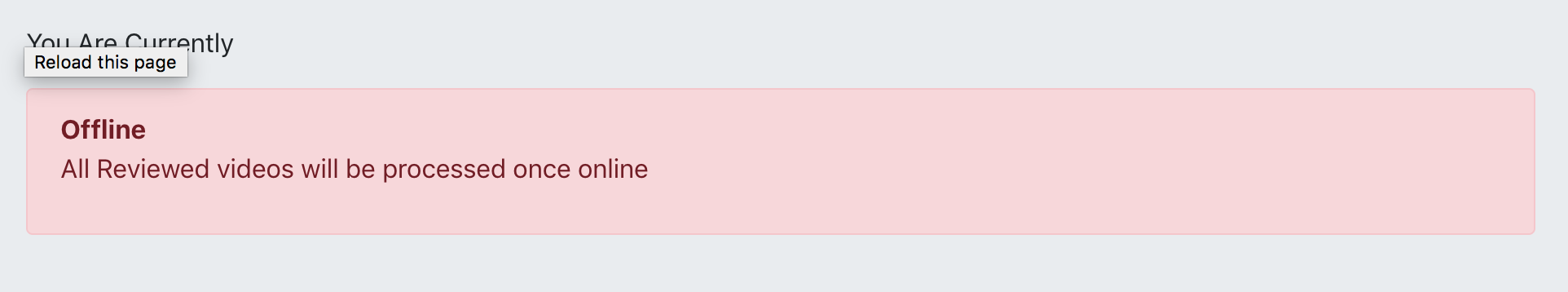
**If the username is entered incorrectly a user message is displayed**

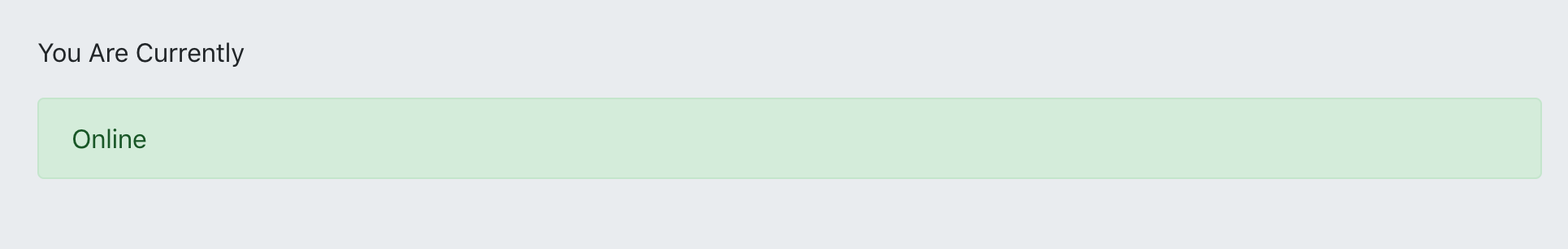


**Dash Screen**

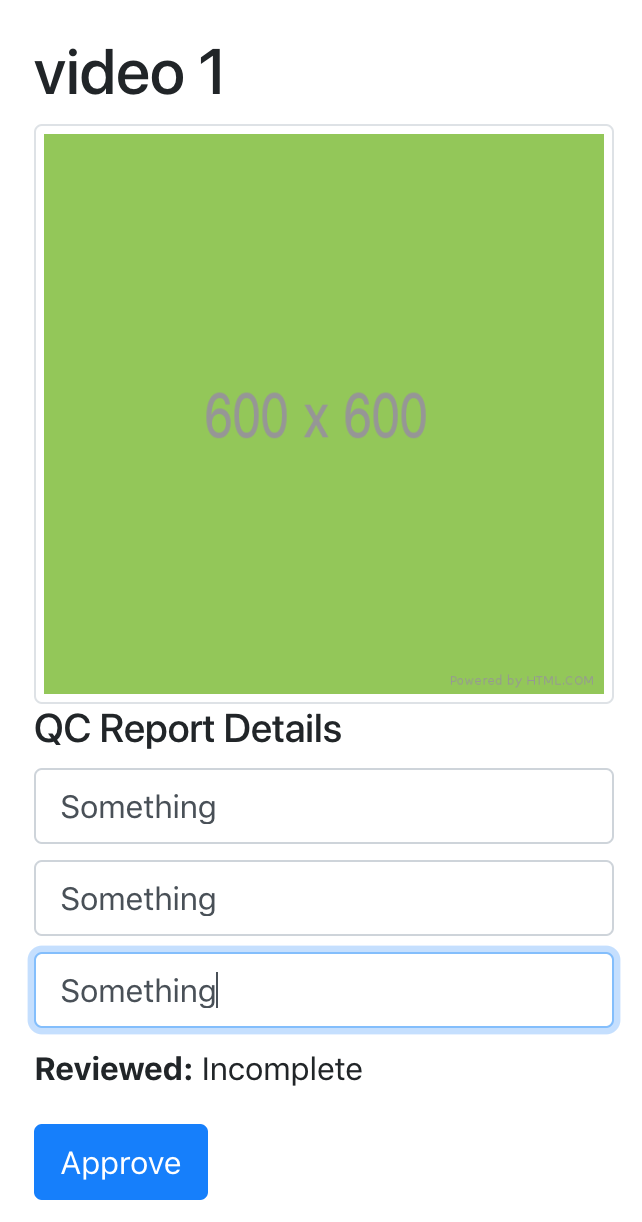
The main dash will appear offline, as this is a state of the application set to false to mimic offline mode. After 5 seconds the system will be online.

To go offline again simply disconnect from the Wi-Fi-connection or go offline in the browsers via the Dev tools under the networks tab.





The videos have been set as dummy data from a json file. The videos will load for the MD in the main dash. A fetch API call is made to get data and render it.



Assumption was made that all the videos that need to be reviewed will be displayed.

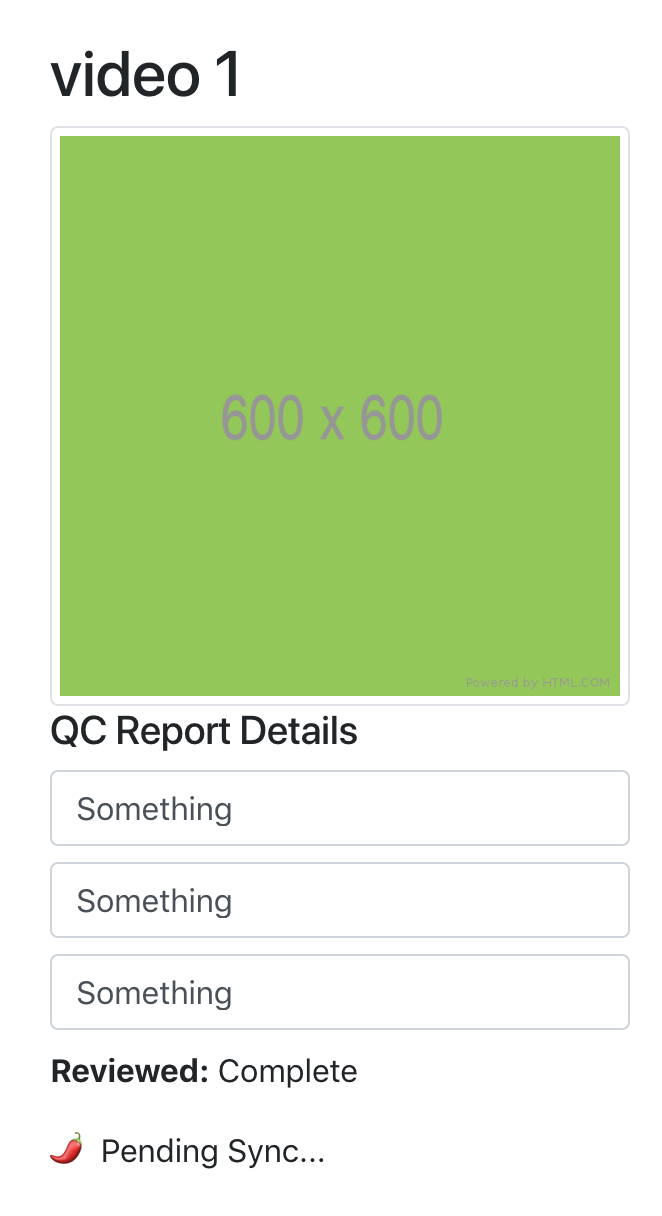
The status Reviewed will be shown as **incomplete**

They MD can fill in the report and then hit ‘Approve’

If the system is online then the item will disappear from the dash

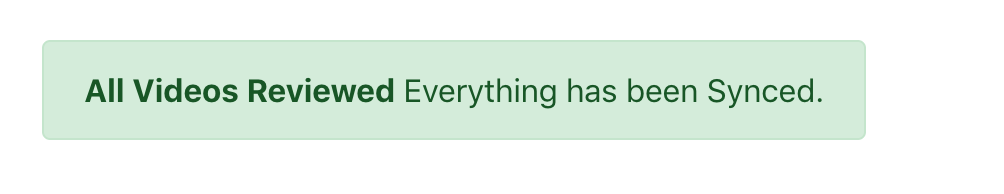
However if the system is offline

The system will go into the status pending sync



Videos in pending sync mode the status will change from **incomplete** to **complete**

**The videos in this state will be queued and when the system comes back on line all items with the status complete will sync and disappear.**



**Conclusion**

This is a prototype of the system which still needs to be refined there are a few bugs however does give the general look and feel of the system beginnings. The reason for choosing React was due to its features however it could have been create in any Frontend library. The State of the application was managed in React no 3rd part library such as flux/Redux was required.