About the company

We aim to bring joy into the lives of our customers through our puzzles!
Our app is called Crossword Star. Every day we release new crossword puzzles with interesting themes.
Till now it's only available for iOS at the moment.

Background

I initially created the app because I thought I could do a better job than what was currently available on the App Store. The other apps felt clunky and didn't feel like a pleasure to use. Having recently retired from teaching, my Mother was looking for a new challenge so I asked if she wanted to create crossword puzzles for this new app idea I had. We launched the app in 2018 and are continually working to improve it.
The server side is programed in PHP and JavaScript. The server uses PHP and the app is also connected to Firebase which uses Typescript/JavaScript for Firebase Functions.

How the server works

Here's how the process roughly works:
1. The server runs on a DigitalOcean droplet which uses APACHE, PHP and MySQL.
2. The crossword puzzles are created on local software and then uploaded to the server.
3. The server saves the crossword and all its information such as title, release date etc.
4. The day before the crossword is released, the crossword is marked as published so that everyone around the world can access the new day's crossword at the start of their day.
5. The app then pulls the crossword data from the server and plays the crossword.

BTW: What language and server set up do you prefer to work with? All of the server code is written in PHP. Thinking long term, I would be open to a rewrite. I also think you'll make a better, more robust API than I have.

The Job: An Admin Website

Overview

We now have over 2000 crosswords and it's getting unwieldy on the local software. At the moment the local software is where everything is set and updated.

I'd like to gradually move the 'hub' online via an admin website. On the site we'd be able to see the crosswords that are coming up, published ones, crosswords that need checking etc.
And gradually we can remove some of the functions from the local software and give the hub more power, such as changing the crossword title, fact, release date etc.

So, at the moment it all goes up to the server.
I'd prefer for it to come down to the local software and only really make changes to the actual crossword via the local software, rather than things like the title and release date.
Does that make sense?

I have created some wireframes so you can have an idea of what I need which you can find here.
I suggest we use the Hyper CSS theme. It will only be seen internally so it doesn't matter too much about the way it looks.

Requirements

1. User accounts
2. View, filter, sort and search crosswords
3. View, filter, sort, search and edit packs

1. User accounts
The site will only be used internally and will need to have user accounts to access it.
Further down the line, different permissions for each user to restrict what they can see and do.

2. View, filter, sort and search crosswords
The site needs to list all the crosswords and make it easy to search and browse through them. See image 1.1 for a wireframe of this page.
When the user clicks on a crossword, they will be taken to the crossword overview page. See image 1.3 for a wireframe of this page.
Clicking on the 'check' button will open a page like this. Our crossword checker currently uses this page to edit and check for errors in the crossword.
Clicking on the 'print' button will open a page like this.

3. View, filter, sort, search and edit packs
The site needs to list all the packs and make it easy to search and browse through them. See image 2.1 for a wireframe of this page.
When the user clicks on a pack, they will be taken to the pack overview page. See image 2.2 for a wireframe of this page.

More information about Packs

Every 1-2 weeks we release a new crossword pack.
A crossword pack consists of a number of crosswords related to the theme of the pack. For example, the Fish pack has crosswords on Salmon, sea fish, Shellfish etc.
Each pack also requires an icon, a hex color, release date.
At the moment I have to manually upload to AWS S3, pick a hex color, and get the epoch timestamp for the release date. Then I have to enter it manually into the database.
It's a cumbersome process so to automate this more, the following needs to happen on the pack overview page.
1. Upload the image and send it to S3.
2. Get the color at the bottom of the pack icon e.g., the blue color at the bottom of this fish icon
3. Choose the release date and time from a calendar in the UI

Changes that may need to be made

The checking functions
At the moment the checking is all done manually. There are no database columns for marking as checked or needs checking or any related info. You'll need to modify the database to include this.

Wrapping Up

It would really help us to start controlling our content from the web. I'm sure you're going to have questions so please let me know.
 Thanks!