**App Code Documentation**

Written by: Chase Dumbacher and Sarah Pham (256-702-7184 or Discord: Sarah #1147)

Spring 2021

This documentation describes the functionality of the app in terms of the code. Explained from our point of view. We will also go over errors and how we resolved them.

When we were first given this project Jan 2021,  
- What was working:

* App/Current Testflight version 1.0.4
* Firebase Realtime Database
* Data from database loading into the app
* Google Drive feature

-Not working/We fixed:

* Emailing system
* Two unused buttons (Bookings and Documents - removed)
* An extra Calendar button (Inside Generate Forms feature)
* Firebase Cloud Functions
* Node version 8 (now 14)
* Color of calendar dots
* Navigation bug where second login would go straight to Venue Lists
* Apple Business Manager/App Store access.
* Updated App Packages/Build

**Code Design Choices**

* React Native was used by original team with website in mind, but future teams code made website and app non-compatible. For this reason, we made a separate app and website.
* Firebase functions containerizes all the code related to sending out documents (invoice, confirmations etc) to allow for debugging and re-use of this code on the website.
* Diagrams visually explaining how certain important functions and design of the code can be found most up to date on our cycle 3 report included in the git

**App Store Version Number**

* It is recommended that every major update on the App, to update the testflight as well, since they expire within 90 days.
* Everytime you upload a new testflight on the Appstore, you have to increment the version number or the build number.
* Syntax = Version Number (Build Number)
* Build numbers are small changes, while the version number is a big change
* Increment the build numbers more than the version number, you will increment the version whenever the build number has exceeded 10.
  + E.g. 1.0.8(1) -> 1.0.8(2)
* This can be changed in **app.json** in the MusicMattersApp folder (top level)
  + “expo”: -> “version”
  + “ios”: -> “buildNumber”
* If you don’t increment these numbers, then the Appstore connect will not accept the archived testflight since there will already be that version number uploaded.

Homepage.js

* File where homepage is

VenueList.js

* File where Venue button leads to
* It has been discovered the VenueViews and VenueViews2 are copied codes with minor changes

EventView.js

* File when clicking on a certain day of the calendar to create a new booking

DocumentationView.js

* File where the Generate Forms button view is
* Uses methods that are within the Index.js folder
* Connects the “ok” button to the methods from index.js (such as genereateSendSaveAll)

Index.js

* Found in functions folder
* Connects the handlers to do the Generate Forms functionality (email, google drive …)

Firebase Functions Env Variables

* Firebase functions stores various variables in an environment file for security purposes.

Bug:

* Login where you double type the first letter for lowercase
  + Solution, make it not case sensitive? FIXED
* Venue titles are looking funky in the calendar view
  + Solution, removed the drop down function and figured out how the venue information was being passed to be able to display the current venue
* Firebase not updating information and Generating Forms crashing
  + Solution, to change the firebase permissions so that everyone logged in could change the database

Fixes:

* Removed Bookings and Document
* Removed random double calendar

Installs:

npm install firebase-admin --save

Notes for Firebase Cloud Functions:

* Before the project was on the Spark plan, which caused a lot of errors within the emailing code because they were using google cloud containers (which you had to pay for)
* Therefore we upgraded to Blaze plan (it was a process with verifying the billing account)
* Now running “firebase deploy --only functions”, we ran into more errors with it not recognizing firebase-admin, I tried everything and running npm install inside the functions folder was NOT the solution.
* The solution was right clicking node\_modules folder (which contains almost everything you need, such as firebase and googleapis and nodemailer) and deleting and then running “npm install”
* Then running the deploy command over and over until you install/fix each issue you run into, because sometimes npm install does not install Cheerio or “googleapis” which you need for this project.
* “firebase serve --only functions”
  + This command in the terminal (in the project directory) will test the functions and see if there are any errors
  + Note: When I ran this, I kept running into errors, especially with the env
  + Error: “Error: Port 5000 is not open on localhost, could not start functions emulator.”
    - Solution: Run “lsof -t -i tcp:5000 | xargs kill”
  + Error: “⚠ Your requested "node" version "10" doesn't match your global version "14" i functions: Watching "/Users/sarahpham/Documents/school/SeniorDesign/functions" for Cloud Functions...
    - Solution: Update Node to 14 in package.json inside the engine object

⚠ It looks like you're trying to access functions.config().drive but there is no value there. You can learn more about setting up config here: <https://firebase.google.com/docs/functions/local-emulator>

⚠ TypeError: Cannot read property 'web' of undefined at Object.<anonymous> (/Users/sarahpham/Documents/school/SeniorDesign/functions/handlers/driveHandler.js:6:61)”

* + - Solution: Run “firebase functions:config:get > .runtimeconfig.json”
    - This manually creates this json file **INSIDE** your functions directory, only run inside that directory
  + Error: GET request is bad
    - <https://github.com/firebase/firebase-tools/issues/2809>
  + NOTE: You need an emulator to use firebase serve
    - Run “firebase emulators:start”
* Functions were all still in Node 8 on Firebase, which was deprecated. So we deleted them and redeployed all the functions so that they could be updated to Node 10.
  + Error: Functions could not be deployed
  + Solution: deploying functions with debug and reading the firebase debug logs on all the GET requests
  + We read that there was an Authentication error while visiting all the GET request links and reading the returns
  + “firebase functions:config:get” returns the login information we have set for the google drive and gmail
  + “firebase functions:config:set gmail.password=”pass”
  + Solution #2:
    - **Npm install --save-exact firebase-functions@[version] npm install --save-exact firebase-admin@[version]**
    - Run above command in functions folder and functions were able to be deployed
* Error:
* Sarahs-MacBook-Pro:functions sarahpham$ node index.js
* {"severity":"WARNING","message":"Warning, FIREBASE\_CONFIG and GCLOUD\_PROJECT environment variables are missing. Initializing firebase-admin will fail"}
* /Users/sarahpham/Documents/school/SeniorDesign/functions/node\_modules/firebase-admin/lib/database/database.js:88
* throw new error\_1.FirebaseDatabaseError({
* ^
* FirebaseError: Can't determine Firebase Database URL.
  + Solution:
    - Graphical user interface, text, application, Teams

      Description automatically generated
    - Passed through the databaseURL
* Error (from above):
  + {"severity":"WARNING","message":"Warning, FIREBASE\_CONFIG and GCLOUD\_PROJECT environment variables are missing. Initializing firebase-admin will fail"}
  + [2021-03-05T01:25:30.291Z] @firebase/database: FIREBASE WARNING: {"code":"app/invalid-credential","message":"Credential implementation provided to initializeApp() via the \"credential\" property failed to fetch a valid Google OAuth2 access token with the following error: \"Error fetching access token: Error while making request: getaddrinfo ENOTFOUND metadata.google.internal. Error code: ENOTFOUND\"."}
  + Solution:
* Google ([musicmattersbookings@gmail.com](mailto:musicmattersbookings@gmail.com)) app password
  + App pass : jjpilshypwifnjac
  + This password is a one-time pass that allows the NodeMailer to use [musicmattersbookings@gmail.com](mailto:musicmattersbookings@gmail.com) without running into 3rd-party auth issues.
  + Password found in the config.env   
    qzvwgnlwqhimhumw

Simulator Bugs (That only show on iOS simulator and not actual testflight device):

* After you login, and you got to the homepage, if you try to swipe backwards, you go to “Create a New Venue” page
* Clicking on a certain event and editing the name for example and then saving and going back does not actually save it
* Editing a venue does not open up the “Edit Venue” page like it would on an actual device

Building/Archiving Issues

* Error: jest-haste-map:` Watchman crawl failed. Retrying once with node crawler.
* Solution: In mac privacy settings, giving every Watchman application and terminal, full disk access

Date/Time Picker UI Bug:

* For reference: [https://github.com/react-native datetimepicker/datetimepicker/issues/285](https://github.com/react-native-datetimepicker/datetimepicker/issues/285)
  + Click the link to see what our date/time picker looked like
* Issue: When we were editing event info, the date/time picker looked funky, which would only show on testflight and not on a simulator
* Solution: Update NPM, update the date-time-picker, update expo cli (expo update)

**Deprecated:**  
npm WARN deprecated topo@2.0.2: This version has been deprecated in accordance with the hapi support policy (hapi.im/support). Please upgrade to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this time, paid support is available for older versions (hapi.im/commercial).

npm WARN deprecated @hapi/pinpoint@2.0.0: Moved to 'npm install @sideway/pinpoint'

npm WARN deprecated @hapi/formula@2.0.0: Moved to 'npm install @sideway/formula'

npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated

npm WARN deprecated har-validator@5.1.5: this library is no longer supported

npm WARN deprecated hoek@4.2.1: This version has been deprecated in accordance with the hapi support policy (hapi.im/support). Please upgrade to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this time, paid support is available for older versions (hapi.im/commercial).

npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated

npm WARN deprecated @hapi/address@4.1.0: Moved to 'npm install @sideway/address'

npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to fsevents 2.

npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies.

npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to fsevents 2.

npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies.

npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142

npm WARN deprecated joi@11.4.0: This version has been deprecated in accordance with the hapi support policy (hapi.im/support). Please upgrade to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this time, paid support is available for older versions (hapi.im/commercial).

npm WARN deprecated @hapi/joi@17.1.1: Switch to 'npm install joi'

npm WARN deprecated core-js@2.6.12: core-js@<3 is no longer maintained and not recommended for usage due to the number of issues. Please, upgrade your dependencies to the actual version of core-js@3.

Deep Dive into “Artist Confirmation” and “Artist Invoice”

* Programmers: Chase and Sarah
* Generate Forms =>
  + The only functionality that worked when we initially started this project was the pdf’s being sent to the drive
  + Artist Confirmations
    - Solution: Redeploying all firebase functions, creating our own test function and calling certain methods within the generateSendSaveOne to test if the bug was inside the code
  + Artist Invoice
    - Note: Invoice is very similar to confirmations - we did not change any of the strings, this is how it was given to us
    - Solution: The bug was in the PDF builder where it was calling venue.address.toUpper(). This caused an error and prevented an invoice from being sent out

“Calendar”

* This initially did not work
* Ways of testing:
  + Taking “generateCalendar” and replace it in a method that we know works, so replacing generateArtistInvoice with generateCalendar.
  + After everytime we changed the code we had to redeploy the firebase cloud functions and wait to test
  + We tested if the Calendar will send to the Drive and the Email
  + We tested if the Calendar will work if we comment out the Email and just have it sent to the Drive
    - Result: The Calendar sent out to the Drive successfully, function did not return an error
  + We deleted all Events from the current month and tested it with the Email/Drive both inside the method
    - Result: We received an empty calendar on the email and the drive, that means the emailer is not where the issue is
  + We added back in some events with Email/Drive both still inside the method (tried both email/drive commented and uncommented, both produced the same result)
    - Result: We received an internal error, so that means that events are the issue, we noticed the generateCalendar was the only one using “events” from the database
  + We found that generateCalendarTable was the only one using “events”, so we looked further into the generation of the calendar, it was a bit confusing at how they were handling events. We put console.log functions everywhere so that it could print on the functions logs when we clicked on the “Monthly Calendar” button
    - Result: All the variables we logged got printed to the firebase cloud functions logs and we were able to
  + **Solution**: The bug was found in generateCalendarTable in the forEach loop. The way the other team was pushing the data into the cellArray, they had “Util.event.start” which was an invalid was of converting the military event.start time to current time.
    - After changing Util.event.start to Util.toAMPM.event.start, the calendar returned a success message!
* You may notice on previous screenshots of the calendar view that the marked events were random colors (such as yellow, green)
  + Mike said he wanted the colors to be more noticeable so we changed the color to a bright #FF000 red on generateMarkedDates()

Emailing List, Sending Emails to Multiple Emails

* You can add multiple emails to a Venue when you edit or create it on the bottom
* Note: the emails are CASE SENSITIVE and will not send if a letter is incorrectly uppercase/lowercase
* Booking List and Calendar uses the emailingList

Editing Venue’s Bug

* We ran into a problem when updating the version to 1.0.7, where if you try to edit a venue, it goes straight to the “Create a New Venue” page instead of “Update Venue” page like it should
  + Debugging: We added console.log and since there are no cloud functions, we had to look at the console logs through React Native’s debugger mode
  + Debugging: After clicking on the button that triggers the console log, we clicked CTRL+D and opened the React Native Debugging website
  + Then right click + Inspect Element and you can view the logs and the errors from console, shows us which line/variable is failing in the code
* Error From Debug:
  + Warning: Failed prop type: The prop ‘onSave’ is marked as required in ‘VenueView’ but its value is ‘undefined’
  + Warning: Failed prop type: The prop ‘database is marked as required in ‘VenueView’ but its value is ‘undefined’
* After debugging a lot, we figured out that after the selected venue is pass from VenueViews2 to VenueViews, the constructor is not being called for VenueViews. This is causing an issue, because the constructor is setting the data parameters to the global variables. VenueViews uses the existence of global variable data to determine whether or not the Venue is in “Edit” or “Create” mode
  + Solution: We put the contents of the Constructor into a testMethod so that we could recall the “Constructor” whenever we were navigating to ‘Venue’
* Edit button now works and pulls up information
* Error: Save/Delete button do not work anymore in the “Edit Venue” page
  + Error Message: Warning: Can't perform a React state update on an unmounted component. This is a no-op, but it indicates a memory leak in your application. To fix, cancel all subscriptions and asynchronous tasks in %s.%s, the componentWillUnmount method
  + Helpful website of component lifecycle in React-Native: <https://projects.wojtekmaj.pl/react-lifecycle-methods-diagram/>
* Error: When trying to type in a new venue name, it keeps calling this render method and it wont let us actually type in anything since we reinitialize everytime
  + Research: Turns out we need to update these “props”
  + ComponentDidMount and ComponentDidUpdate seem interesting
    - <https://reactjs.org/docs/react-component.html#componentdidmount>
* Solution
  + I (Sarah), was first working on solutions involving the componentdidmount and componentdidupdate
  + Chase, was working on a “this.props.navigation.addListener” so that the props would know when to update the state of the variables
  + Somehow, our combined methods became a solution
  + Now we can type in “Edit Venues” without it reinitializing
  + For some reason, the listener is being called on the component update - Chase
  + Received this warning: ExceptionsManager.js:126 Warning: Expected VenueView state to match memoized state before processing the update queue. This might either be because of a bug in React, or because a component reassigns its own `this.props`. Please file an issue.
    - Lets ignore it for now -Sarah
* Received this warning when saving:
  + ExceptionsManager.js:126 Warning: Can't perform a React state update on an unmounted component. This is a no-op, but it indicates a memory leak in your application. To fix, cancel all subscriptions and asynchronous tasks in the componentWillUnmount method.

“Export to Excel”

* You may notice that previous screenshots of the “Generate Forms” feature will show an “Export to Excel” functionality.
* A meeting with Mike on a cloudy Saturday (March 20, 2021) ended with the conclusion that the feature was unnecessary.
* So it was deleted.

Things We Refactored

* Removed duplicate pdf creation
* Compress email handler and gensend documents functions
  + GenSendSaveInvoice and GenSendSaveConfirmation -> GenSendSaveInvoiceOrConfirmation