

Hitta

BS.c. Tölvunarfræði
T-404-LOKA

Rekstrarhandbók

Höfundar:

Davíð Fannar Ragnarsson

Matthías Finnur Vignisson

Stefán Örn Ómarsson

Leiðbeinandi:

Ásgeir Freyr Kristinsson

Prófdómari:

Þorgeir Ómarsson

13. maí 2021

Table of contents

1. Overview of system	3
1.1 General	3
1.2. Naming Convention	3
1.3. Git	3
2. Database	4
3. Setting up software	4
4. Information on setup for Windows	4
5. Information on setup for OSX	5
6. Setup of development environment	5
7. Setup of dependencies for the project	5
8. How to examine and test the project	6
9. Maintenance	7

1. Overview of system

1.1 General

All code is written in React-Native which mixes together the best benefits of React and certain JavaScript libraries to develop a user friendly UI for IOS and Android, our project Hitta focuses on development for IOS and all code is written in Visual Studio Code.

1.2. Naming Convention

To have understandable and well written code we need strict descriptive and clear naming conventions instead of non descriptive or unclear names. Names of variables and functions are written in [Pascal-casing](#), for example, “deleteEvent” or “getTimeLeft”. When functions or variables need more explanation than is readily apparent we comment on the code. All code is written and commented on in english.

1.3. Git

All code is stored in the version control system on GitHub which can be found here:

<https://github.com/davidfannar98/Hitta.git>

All code is stored in the branch “main” and most code has been pushed there. In working on the project in React-Native we have learned that there are not many Merge conflicts as we have organized it so that people are generally not working on anything that can cause conflicts without alerting people prior. In the rare case there was a conflict it was easily read through the code and changed manually.

2. Database

We employ several services from Firebase:

Firebase Authentication: Stores information on users(name, email, password) in a secure manner.

Firebase Firestore: Stores information on everything about everything we need to store for events aside from images.

Firebase Storage: Stores all images related to events.

Here you can take a closer look at firebase and their services:

<https://firebase.google.com/>

3. Setting up software

Before it is possible to look or test the project, specific programs need to be in place to ensure it runs correctly on the system. The following information relates to how to set up this environment.

4. Information on setup for Windows

1. Check if Node.js is already installed by running the command:

```
node -v
```

2. If Node.js is installed it will then return for example "V14.15.5". To upgrade to the latest version you can run:

```
npm install -g
```

3. If Node.js is not installed, you can then get it from <https://nodejs.org/en/download/> and select the suitable OS for installation.

4. By setting up Node.js you have also set up npm. It is best to get the latest version of it by running:

```
npm install npm@latest -g
```

5. Next you need to check if Git is installed by running the command

```
git version
```

6. If Git is set up you will receive for example "git version 2.29.2.windows.2". Otherwise you will need to download git for your OS here: <https://git-scm.com/download/win>

5. Information on setup for OSX

1. Check if Node.js is already installed by running the command:
`node -v`
2. If Node.js is not installed, you can then get it from <https://nodejs.org/en/download/> and select the suitable OS for installation.
3. It is best to update npm to make certain you have the latest version.
`sudo npm install npm --global`
4. If Node.js is installed it will then return for example "V14.15.5". To upgrade to the latest version, a package manager like n is best to do so, you can get it and get the latest version by running the following commands:
`sudo npm install -g n`
`sudo n latest`
5. Next you need to check if Git is installed by running the command
`git version`
6. If Git is set up you will receive for example "git version 2.29.2.osx.2". Otherwise you will need to download git for your os here: <https://git-scm.com/download/>

6. Setup of development environment

To work on and test Hitta you will need Expo CLI, which can be set up with the commands:

Windows : `npm install --global expo-cli`

Osx : `sudo npm install --global expo-cli`

7. Setup of dependencies for the project

To ensure that Hitta functions correctly you will need to install the required dependencies to the system which are all stored in "package.json". You can install them by running the command in the root folder of the project:

```
npm install
```

8. How to examine and test the project

To run the project you will need to download the program Expo Go onto your IOS device. Then in the IDE terminal you type in:

```
expo start
```

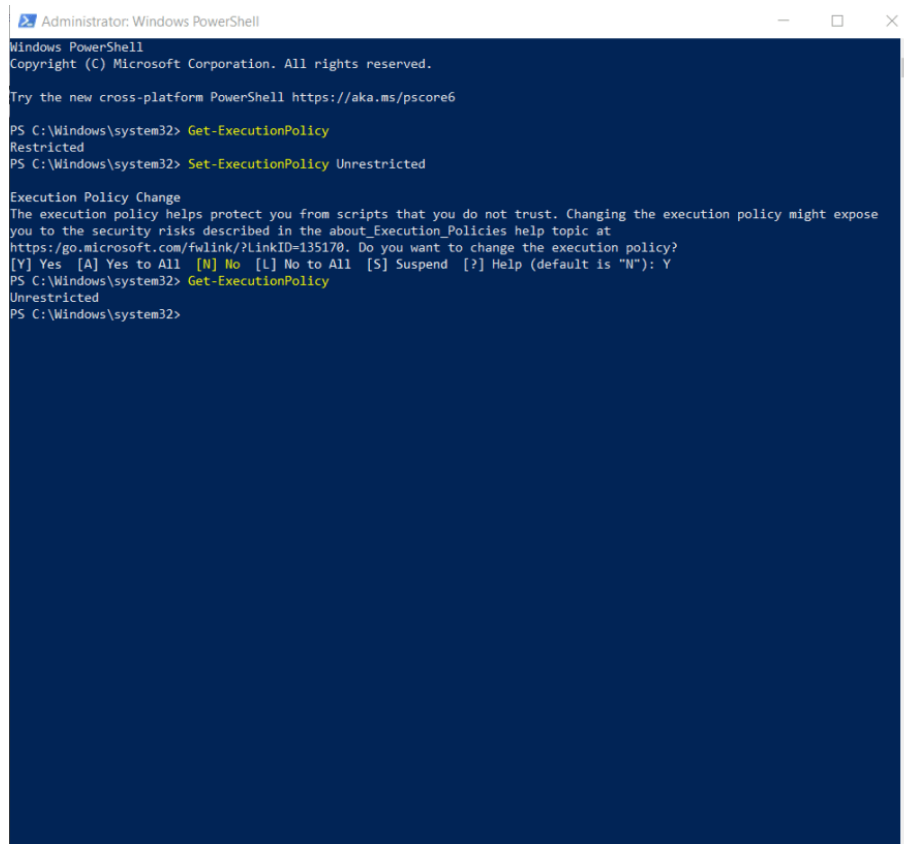
That will start the program. You will then need to either scan the QR code that comes up with your camera, send a link via email to your device, or run on a simulator on your system if you have one setup. The system will then open up on the device and you can test all its functions. Your mobile device needs to be on the same local network as your computer. We recommend using iPhone X or newer versions since some pages are not responsive enough to handle smaller screens.

If you get this error running “expo start”:

cannot be loaded because running scripts is disabled on this system. For more information, see [about_Execution_Policies](https://go.microsoft.com/fwlink/?LinkID=135170) at <https://go.microsoft.com/fwlink/?LinkID=135170>.

You need to open *powershell* as an administrator and enter “Get-ExecutionPolicy”.

If it says *restricted* you need to enter “Set-ExecutionPolicy Unrestricted” and type in “A” to select [Yes to all]. Now you should be able to run expo by typing expo start in the terminal.



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> Get-ExecutionPolicy
Restricted
PS C:\Windows\system32> Set-ExecutionPolicy Unrestricted

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose
you to the security risks described in the about_Execution_Policies help topic at
https://go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): Y
PS C:\Windows\system32> Get-ExecutionPolicy
Unrestricted
PS C:\Windows\system32>
```



9. Maintenance

If Hitta lives for a longer time then all the dependencies that can be found in “package.json” will need to be updated when deprecated. That can be done by going to the respective library(dependency) webpage to see how to update to the newest version.

Our database management system Firebase is free of use as we’re using it now. But in order to scale it up to allow large numbers of users we would need to upgrade it to Firebase Blaze plan which is not free of use.

And above all keep our code up to date to the latest standards in the extremely fast evolving world of computers and computer programming.