12/2/2018

Login and Register application with weather viewer, developing a mobile application with the back-end developed in the previous course work

Course work 2

WAEL ALDROUBI

Table of Content:

|  |  |
| --- | --- |
| PAGE | CONTENT |
| PAGE 0 | Cover page |
| PAGE 1 | Table of content |
| PAGE 2 | Introduction & Explanation of the code |
| PAGE 3 & 4 & 5 | (Front-end) Client side Explanation |
| PAGE 6 & 7 | (Back-end) Server side Explanation |
| PAGE 8 | Conclusion |
| PAGE 9 | References |

**Introduction:**

I am required to develop an mobile application that have my previous back-end and to develop the security in my application depending on my research about web applications security, and this will be my first part of this course work, then developing a mobile app that provide some services, so I have login, registration and then the second activity will be live weather viewer with logout button.

My back-end is depending on JavaScript frameworks like express, node js, mango dB, and angular js, then I had developed some security codes, for my front-end I had added some methods to improve the security like a counter for wrong tries, and then sending email verification to new registered users to activate their account before login.

In the following I will explain the whole code with security methods used and will provide screen shots.

**Explanation of the code:**

My code is divided into two parts, mobile front-end (client side) and application back-end (server side).

**Mobile front-end:**

I have three activates, 1st one is login activity called Main activity, and the design consists of username and password text fields and then a login button with a counter to five for wrong entries then will block access to login button in case anyone trying with a friend account, and a button for registration if the user has no account yet.

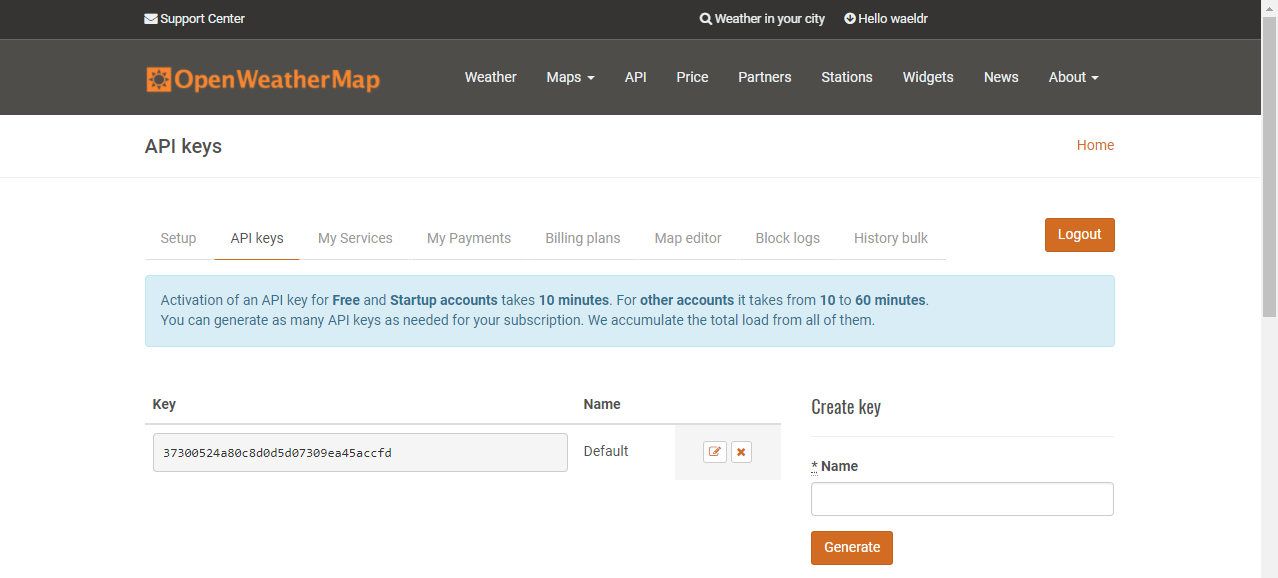
Second activity is registration which is consisting of username, email and password text fields and a registration button and a button for already registered users which will lead the Main activity login page.

Third activity comes after login will lead to my weather viewer application and will show live weather broadcast from JSON file using Openweathermap.com APIs.

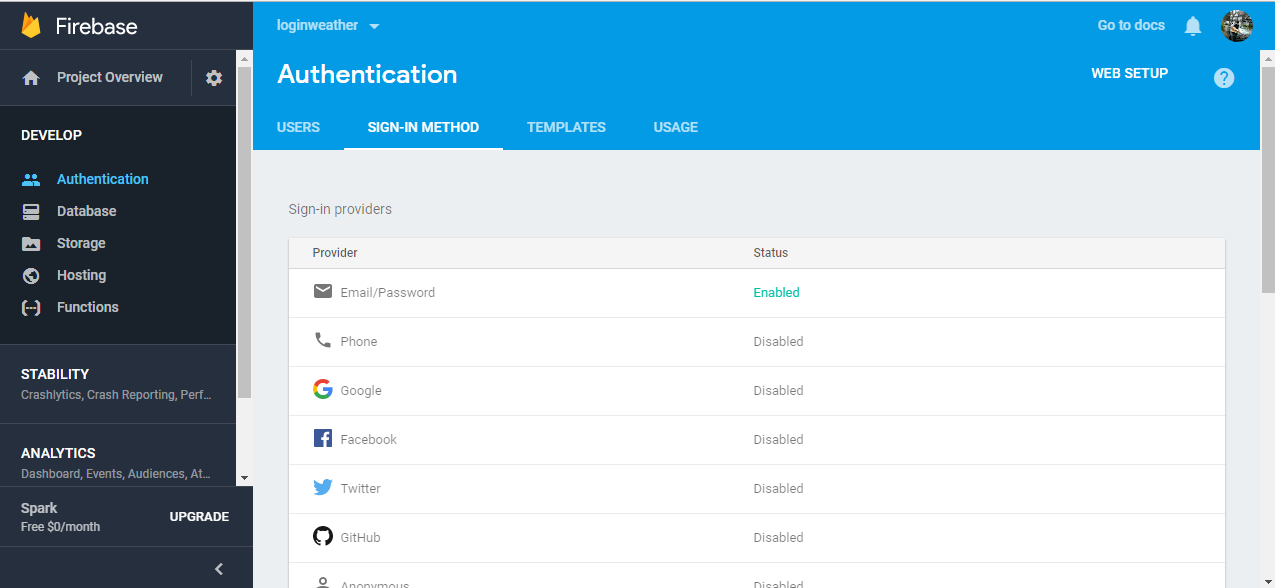
**Mobile front-end security:**

1. A counter to five of wrong entries then will block or deactivate login button, and the only option left is to register.
2. My mobile app is connected to google firebase authentication service and database, so new registered users will receive a verification email to activate their emails before login.

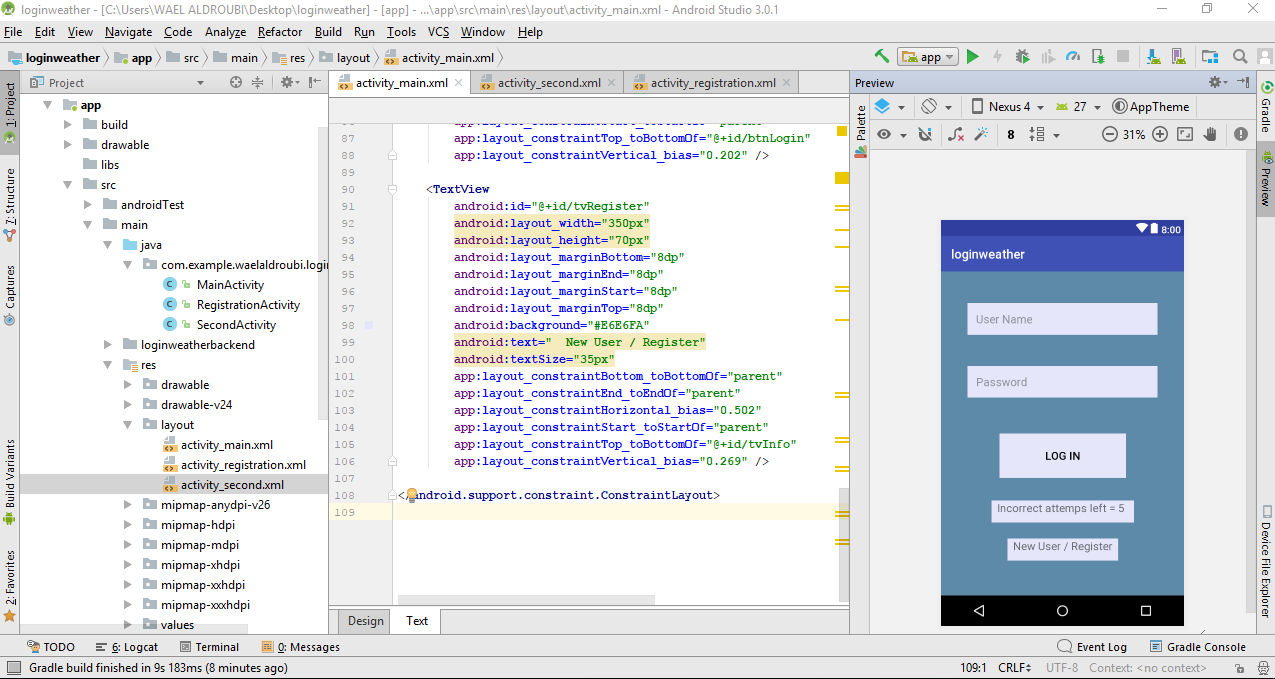
**Client side:**



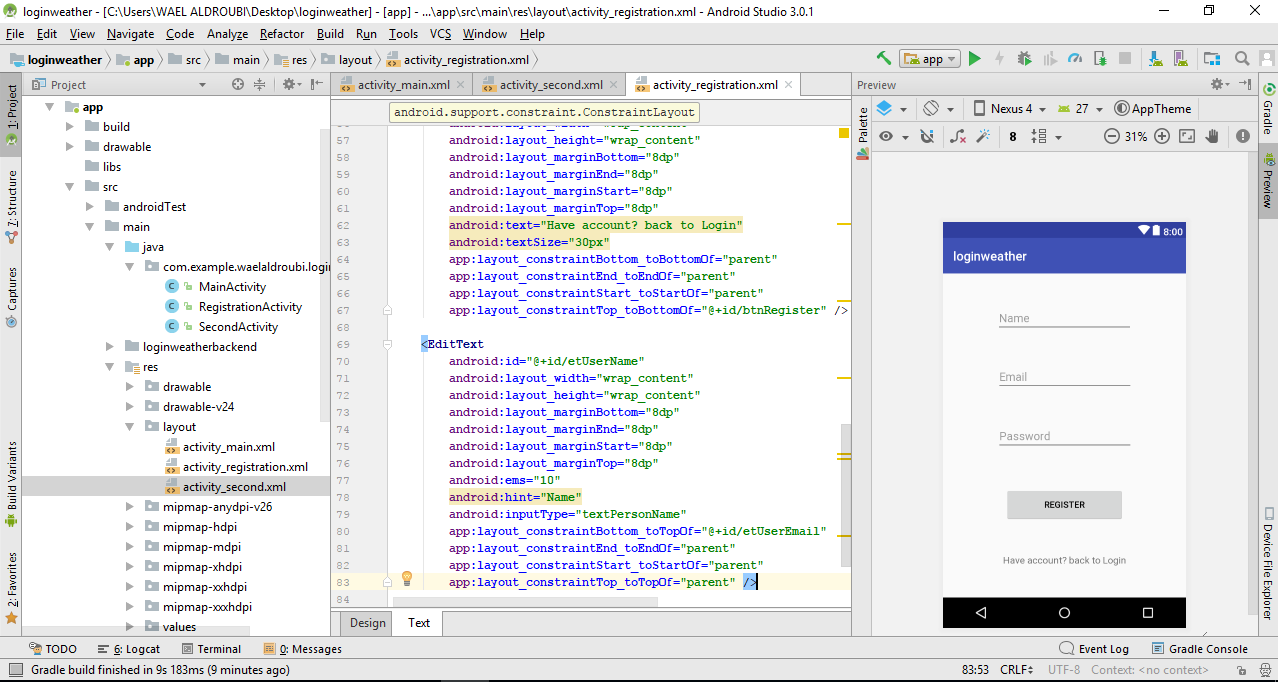
Using openweathermap.com to get live weather in my application



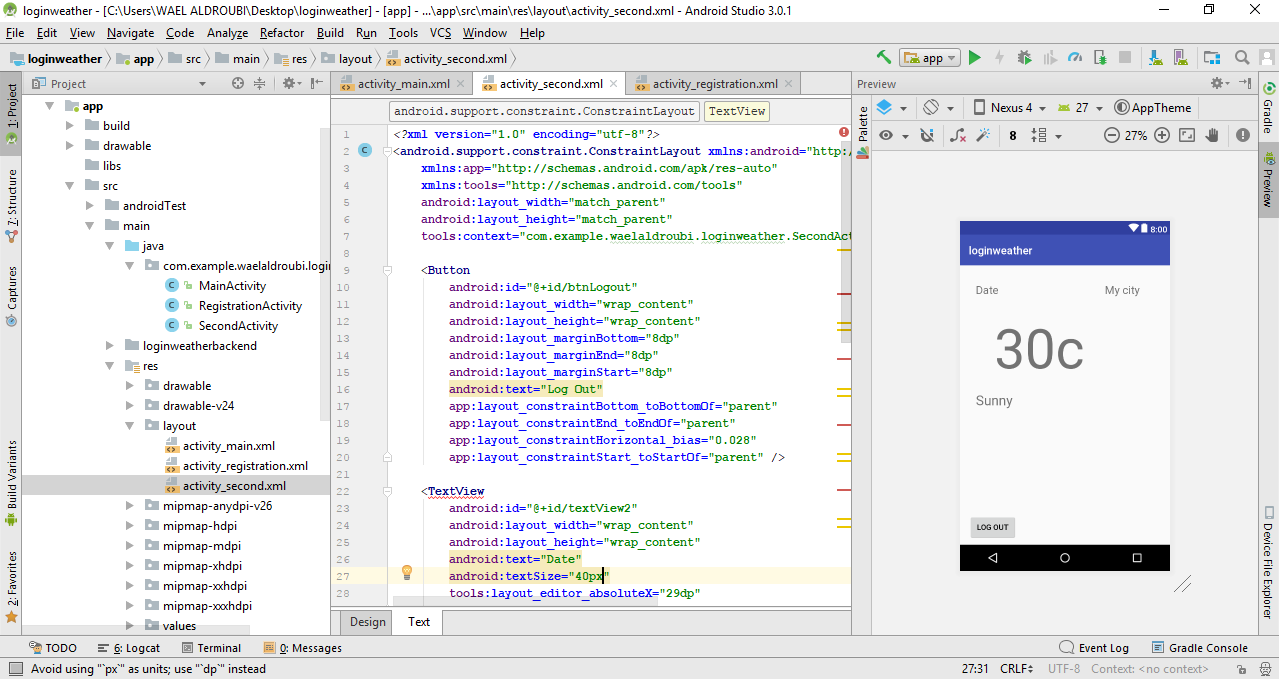
Using Firebase google database and authentication for new registered user to receive activation emails before login to application.



1st activity, Main activity for login page.



2nd activity, registration activity for users to register new user.

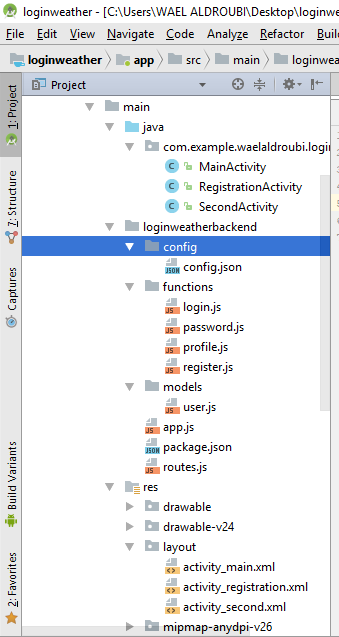


3rd activity to show my live weather application with logout button to go back to login page.

**Back-end code explanation (server side):**

Functionality and security added:

1. Basic-auth: the username and password sent to user as Base64 as a string in header then it is decoded by this module.
2. Bcrypt: Passwords are hashed using Bcrypt hashing algorithm.
3. Body-parser: used to parse the JSOB body in each request.
4. Express: framework used to create RESTful web services.
5. JSON web token: used for session handling.
6. Mangoose: to connect to mango db database.
7. Node mailer: used to send email using SMTP for users who forgot their password.
8. Random string: to generate random string, used as token for rest password.
9. ES6: using ES6 syntax and Promises instead of call backs.



**Code structure:**

1. **Config.json**: to hold name, email, password for creating JSON web token, and will be used to send STMP email.
2. **User.js**: we define our mango db in it, instead of using var we are suing ES6 const and let identifiers for assigning variables.
3. **Functions**: I have four functions, register, login, profile and password.  
   a) **Register**: saves data to mango db, using ES6 promises instead of normal JavaScript call backs, resolve function for success tries and reject function for failure ones.  
   b) **Login**: where authentication is places and then return JSON web token or throw an error.  
   c) **Profile**: has function to return users profile.  
   d) **Password**: where the functions to change password and rest password are placed, to rest password we have two steps, 1st one to generate a random token and send it to user using node mailer and the 2nd step is to verify the token and change user’s password.
4. **Routes.js**: is where RESTful is defined, using express to route, and standard authentication end point is also included, used to add new users, user profile end point is also included and it is where JSON web token is present to get users information like username and password, changing user’s password with new one in case of forgetting it is present in put function with JSON web token, and to rest password, a token for two minutes is generated and sent to user’s email.
5. **App**: is where node js starts, to start listening to the server and we use body-parser, morgan and express here.

\*) Before running the application we need to run services so we write in the terminal:  
1) npm install: it will install all packages in package.json.  
2) npm start: it will lead to app.js automatically.

**Conclusion**:  
Developing web application and mobile applications changed the IT world around us, on daily biases it is growing and becoming more important for users, developers are working hard to create new frameworks and companies are introducing new technologies every day, making web application or mobile application by one person is hard and time consuming, while doing it as a project and manage is in a life cycle with a crew of developers ,designers and a project manager will be the best implementation of a successful web application or mobile application or both of them.

Security is a big matter now after the cold war and all the attacks of mature programmers out there or professional ones, there are many ways to protect your application, and many ready to use frameworks to use, and it will increase the chance to have a strong built application.

Programming manually is possible but having all those backend frameworks makes life easier and faster, the amount of coding is much less than the normal way with PHP, all other functionality is included in the packages, with a huge collection of services, using firebase gave me more security with authentication and verifying users and managing users in a database.

Almost every big company have a web application and mobile apps as well, and some of them have more than one, it is a great way to enter the market and even new small companies are making application at the startup, even before be in the market, as they are the fastest way to promote and to reach users and customers.

# References

Amal, R. (2016, Sep 24). *learn2crack*. Retrieved from learn2crack: https://www.learn2crack.com/2016/09/android-user-registration-login-node-server.html

DK, P. (2017, Jul 13). *youtube*. Retrieved from youtube: https://www.youtube.com/watch?v=lF5m4o\_CuNg&index=1&list=PLbte\_tgDKVWQOCRIzkgEQ8umdn\_S6ZnHr

Experts, P. (2017, Sep 22). *youtube*. Retrieved from youtube: https://www.youtube.com/watch?v=8-7Ip6xum6E&index=28&list=WL

google. (n.d.). *firebase*. Retrieved from firebase: https://firebase.google.com/

openweathermap. (n.d.). *openweathermap*. Retrieved from openweathermap: https://openweathermap.org/