An Online Web-Streaming Service for Bitcoin-Exchanges

Individual Report

Information Technology Capstone Project
COMP5703

Group 28
Zhiliang Wang (460094203)

School of Information Technologies
Faculty of Engineering and Information Technologies
The University of Sydney
2018

Table of Contents

1. Roles and Responsibilities

Our project is to build a web application for collecting and visualizing data from cryptocurrency trading among different exchange websites. The application could visualize real-time data based on the real-time trading running on the global cryptocurrency market. For me, I will be responsible for part of job in front-end programming (including sign up and login modules, materialize.js framework and map bubble in amchart.js) and unit testing.

2. Individual Achievements

First of all, I have found out a nice web deployment tool, named web server for chrome, which can be used for setting up Web server without any complex operations. It will save us so much time working on webserver like apache2.

Figure 1 How to use web server for chrome.

Secondly, I will do some part of front-end programming in the second stage. I will code on sign in and login modules (As shown in figure 2). This will help our users save historical trading data and some important and personal alerting for bitcoin exchanges information.

Figure 2 Sign up and login modules

The user will give some part of authentation to our web server. After that our developers can get access to their personal trading data (As shown in figure 3). I will cooperate with my team mate "Jiyang Li" to complete the logic design and data format job in this part.
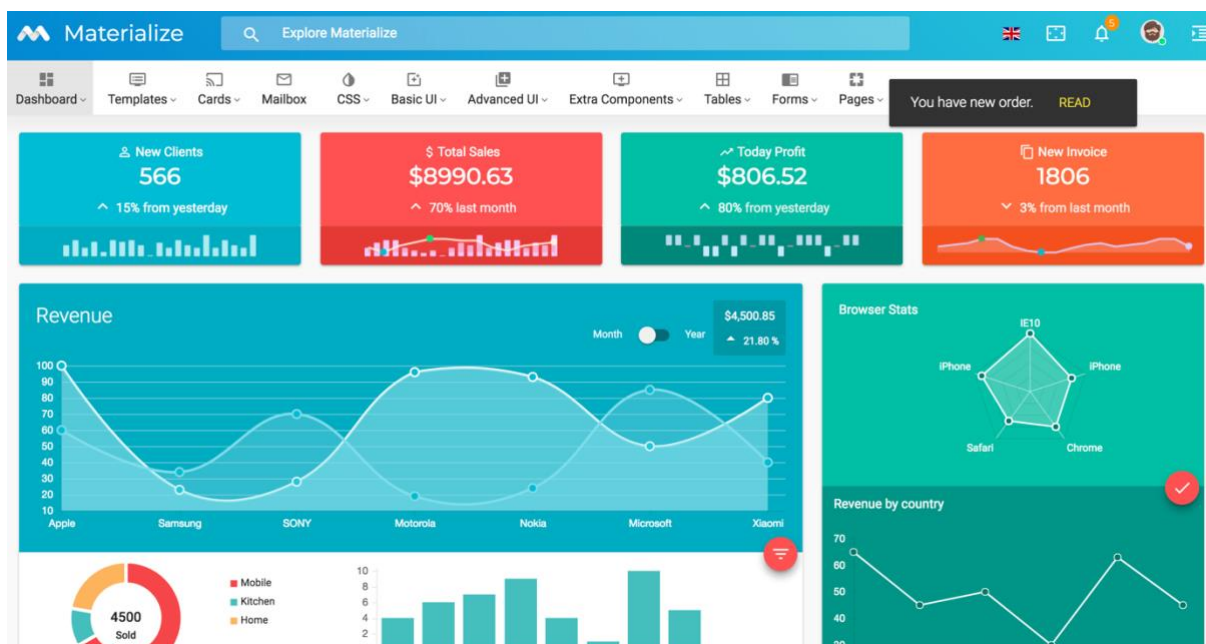

Figure 3 Some of charts and functionality modules on the website.

Through real-time data obtaining process and back-end caculation, I will use amchart.js to show real-time ratio of trading frequency all the world (As shown in figure 4).
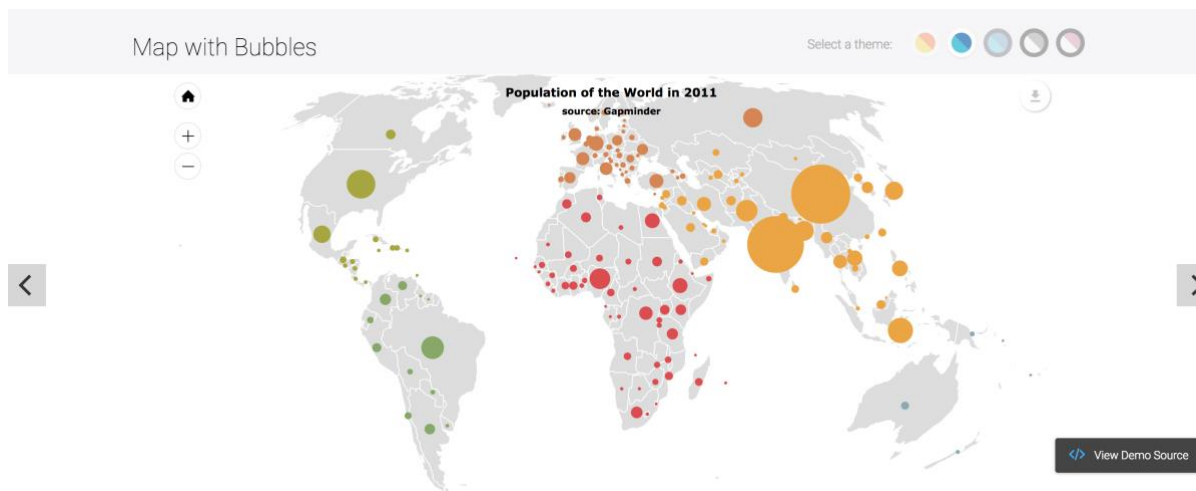
Figure 4 Map with bubbles in amchart.js.

Thirdly, I will be responsible for web application testing which includes functionality testing, usability testing, compatibility testing, performance testing, security testing.

3. Group Collaboration

Firstly, I am an information digger in our group. For example, sometimes I would like to share some good blog or technical essays for team members, which would give some useful advices from these articles(As shown in figure 5&6).
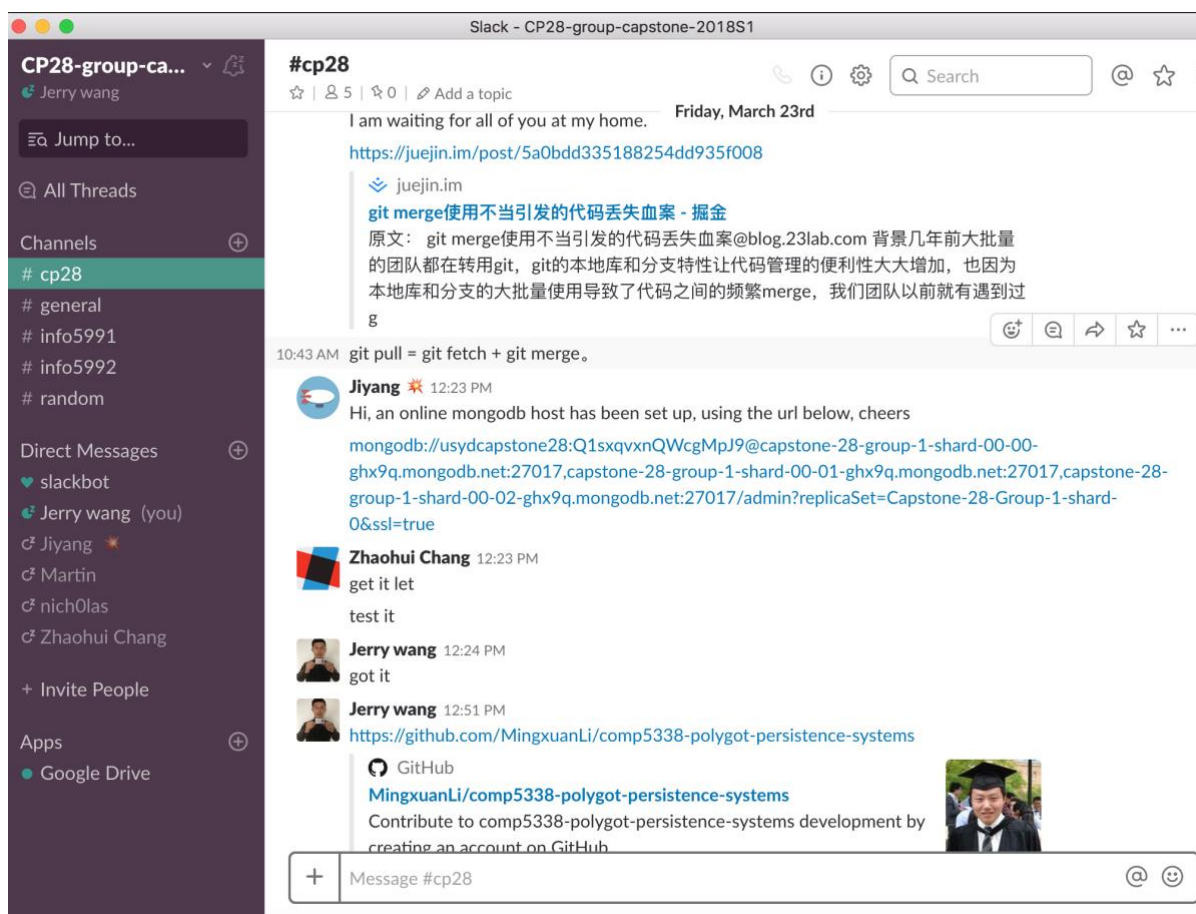
Figure 5 How to use "git pull" correctly and share a github project on MongoDB.
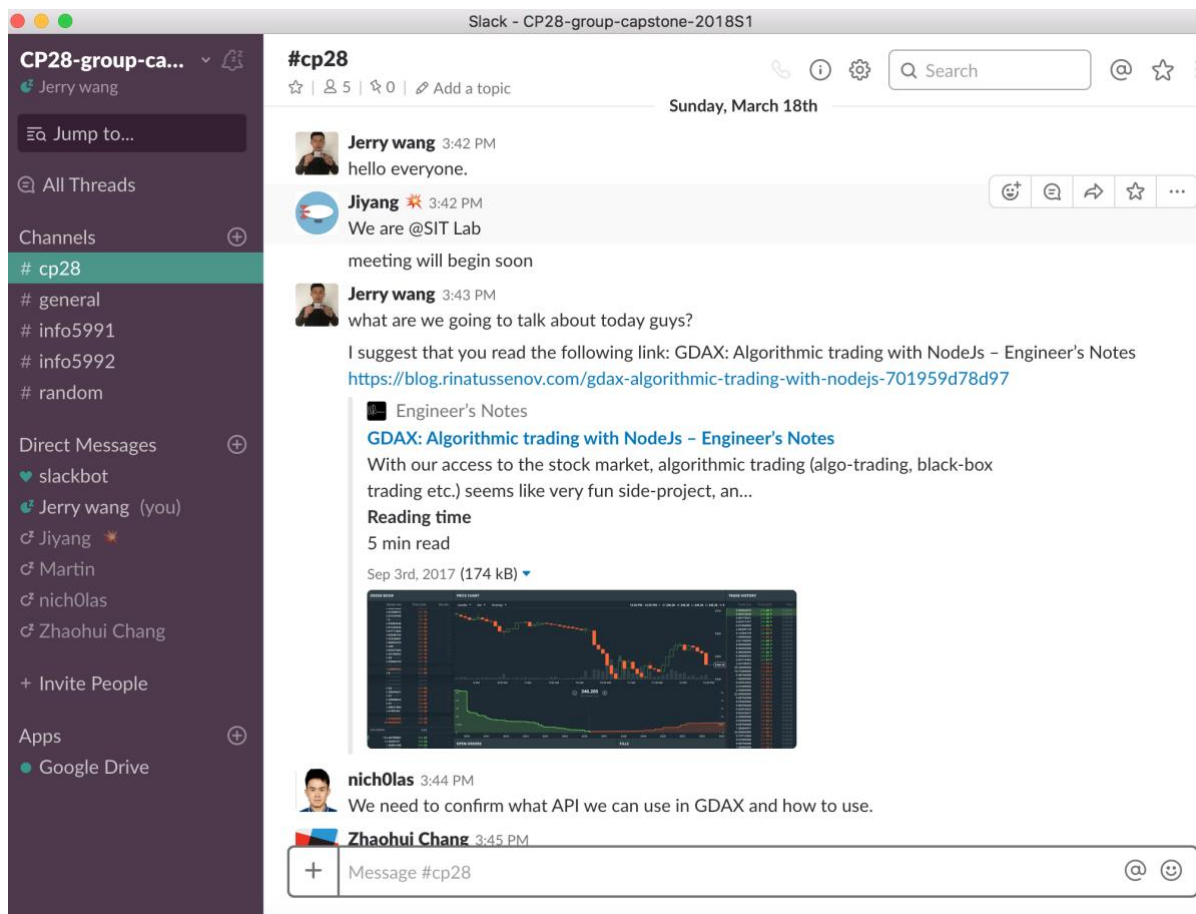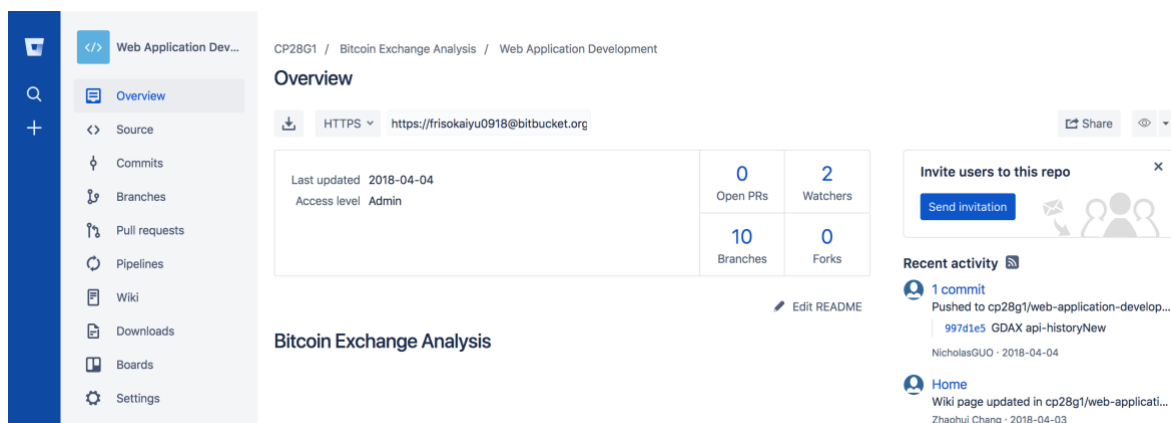

Figure 6 A web note talked about "GDAX: Algorithmic trading with NodeJs".

Secondly, as a group, we will share our latest code to bitbucket which is convenient for us to communicate with coding issues, especially wiki functionality in bitbucket (As shown in figure 7). The address is https://bitbucket.org/cp28g1/web-application-development.

Figure 7 Our project and my commits on bitbucket.

Our project timeline is separated into two lines. The first timeline is from the very beginning to mid-semester and the second timeline is from mid-semester to final presentation. Furthermore, most backend task will be finished in the first timeline and most front-end job will be completed in the second timeline.