Silicon Valley Code Camp 2017 (CMPE 202)

Name: Anup Kulkarni

Student ID: 011772475

Email Address: anupvijayakumar.kulkarni@sjsu.edu

The Node.js Event Loop: Not So Single Threaded

https://www.siliconvalley-codecamp.com/Session/2017/the-nodejs-event-loop-not-so-single-threaded

The talk was very informative. The speaker started with speaking about multitasking the reason it was created and then speaker spoke about limitations in multitasking and how preemptive multitasking is created and finally symmetric Multi-threading(SMT). Really great low-level explanation. And then speaker went to speak that, JavaScript isn't the only code in Node.js. There's also a lot of C++, most important of which is the event loop. One of the event loops responsibilities is to manage a set of threads called the thread pool. As it turns out, most C++ code in Node.js is multithreaded. The session was conducted in Town Square B.

How to write code others wont hate
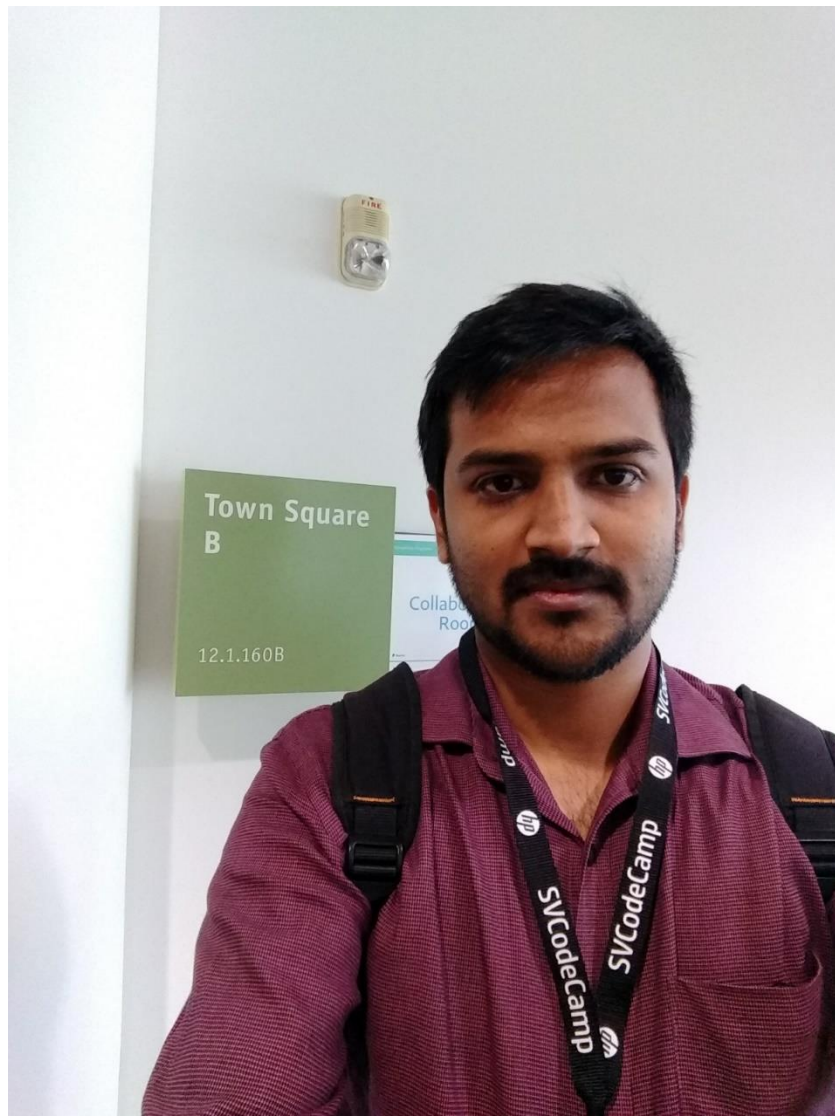
https://www.siliconvalley-codecamp.com/Session/2017/how-to-write-code-others-wont-hate

The speaker started off with a short story that how he got a call from his CEO during a Sunday afternoon in middle of barbeque party when a major bug found in a module of an application. Speaker mentioned that when he started analyzing the code written by someone else he couldn't find proper code comment, there were no proper commit message in git either. This lead speaker to conduct session on how to write code others won't hate. Session was informative and fun. In this session Speaker shared good practices that needs to be incorporated while coding. This session was also conducted in Town Square B.

Speaker mentioned following points to make sure others won't hate our code:

1. Writing proper comments: write comments that cares for why instead of what?
2. Commit history of git hub: Good Commit messages in git hub
3. Code Naming: Get() retrieve() fetch() set() save() post()proper coding convention.
4. Devops can't make changes when code is hardcoded by developers. Therefore, usage of properties file is needed. Critical code paths may be controlled in properties file
   All locations user names etc belong in property files.
5. Add health endpoints
6. Code duplication
7. System features:
         Abstract – use design patterns
   User Facing features:
         Concreate:

How to maintain clean code across the team:

1. Code review system
2. Create pull request 3.
3. Code review
4. Every line of code reviewed by two developers
5. Take random component and try to implement features to check if the component function

Silicon Valley Code Camp 2017 (CMPE 202)

Rock Your Code Quality
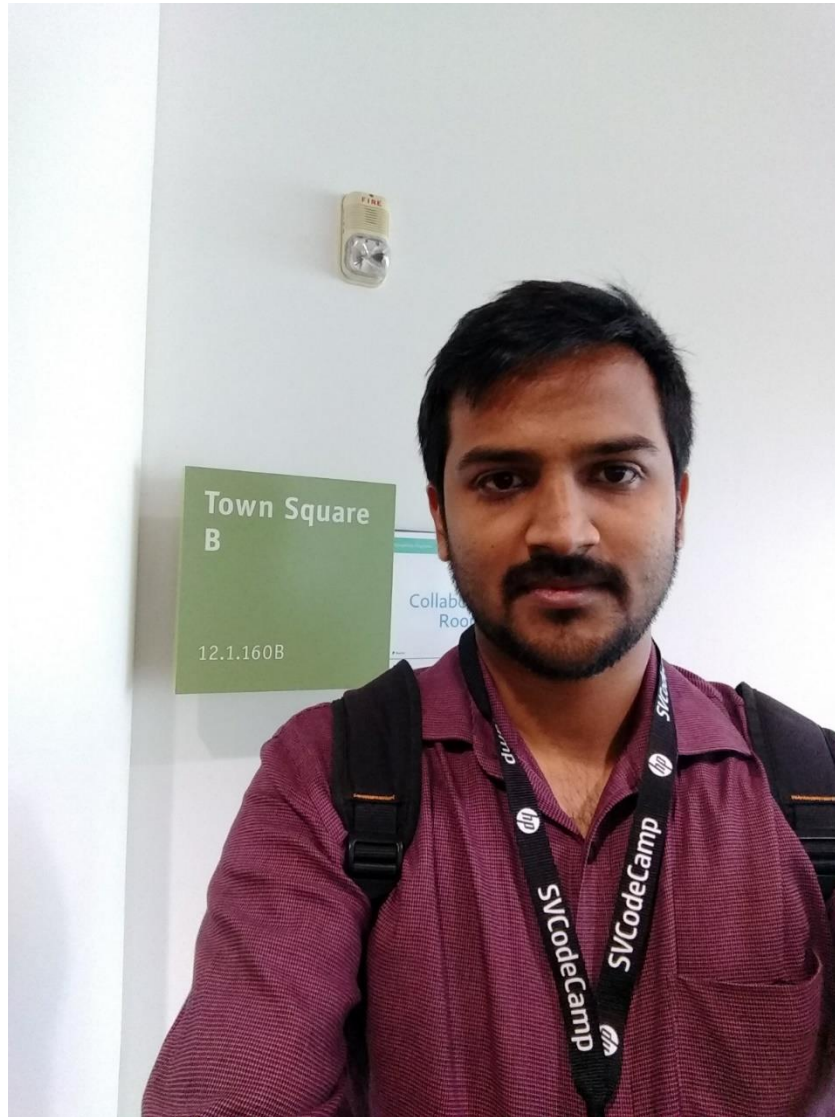
https://www.siliconvalley-codecamp.com/Session/2017/rock-your-code-quality

This session really rocked! Speaker engaged audience starting a video presentation and then giving a brief introduction to rock the quality of the code! Speaker based his talk by asking questions like Why do we need code quality? How to implement code quality easily? And how-to refractor the code to increase the code readability.

Defined code quality in one word: Clarity, Stable, Simple, Undervalued, Maintainable, Tested, SOLID principles.

The Cost of fixing bugs increases as the bugs go through different phases.

1. Requirements – 1
2. Design – 5X
3. Code – 10X
4. Dev testing – 20X
5. Acceptance – 50X
6. Production > 150X

How to implement code quality easily and at different layer?

- Write Coding standards and enforce it on the developers is very important.
- Memory, Network – release objects as soon as possible and check network resources
- Security – SQL Injection, Identity, cross site scripting, Type Checking (JS)
- Reuse – 90% of code should be in DLL's, Portable libraries
- Type Coupling – Tight coupling us bad but loose coupling good.
- Error and exception handling using try/catch – Keep the cpde try catch
- Use global trapping
- Log,log,log - log in many different places.
- Save the code to cloud
- Queuing – writing apps such a way they are not connected to internet (minimal functionality like a tweet or ability to post a facebook post)

Refractor the code using - https://www.devexpress.com/products/coderush/

Speaker also quoted Martin Folwer's quotes that says:

1.I find that writing unit tests actually increases my programming speed.

2. Any fool can write code that a computer can understand. Good programmers write code that humans can understand.

Silicon Valley Code Camp 2017 (CMPE 202)

Silicon Valley Code Camp 2017 (CMPE 202)

Some more picture from the event:

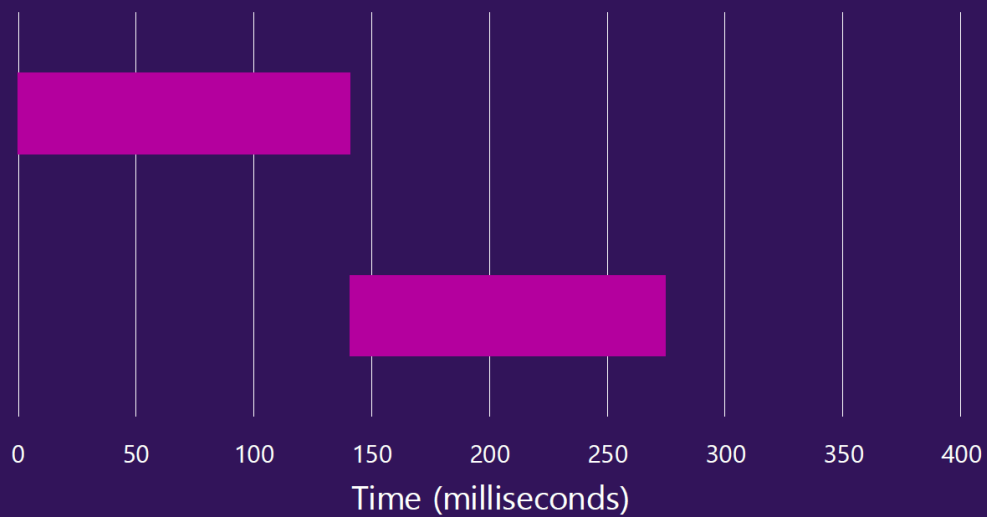## Silicon Valley Code Camp 2017 (CMPE 202)

Here is a code to demonstrate the synchronous and asynchronous behavior explained to use node concept:

Synchronous Crytpo Code:

```
const crypto = require('crypto');
const NUM_REQUESTS = 2;
for (let i = 0; i < NUM_REQUESTS; i++) {
crypto.pbkdf2Sync('secret', 'salt', 10000, 512, 'sha512');
}
```

Implementing the asynchronous crypto code:

```
const crypto = require('crypto');
const NUM_REQUESTS = 2;
for (let i = 0; i < NUM_REQUESTS; i++) {
crypto.pbkdf2('secret', 'salt', 10000, 512, 'sha512', () => {});
}
```


Asynchronous Crypto Results, 2 requests — Execution Timeline — Time (milliseconds)