4/25 - 5/07

TEAM: GIT CONFUSED SPRINT 7

PRESENTER/PRODUCT OWNER – BOB

SCRUM MASTER – EZEQUIEL

DEV TEAM – CHRIS

DEV TEAM – GIL

DEV TEAM – DOM

DEV TEAM – ZACK

DEV TEAM – MARK

USER STORIES

As a user of JVPM:

- I want to have current code functionality is in place.
- I want to have test coverage of 95% or better
- I want to have the code cleaned up
- I want to have files closed when open
- I want to have a requirements file to pull dependencies

STORY POINTS

Cleaning up sniffs on soundcloud. (3)

Remove commented code. (1)

Collect requirements to run the software. Put them in a format that is easy to follow such as a requirements.txt file. (1)

Use 'with' statements for opening files so that the files are closed after use. (2)

Remove '# pragma: no cover' from all tests. (1)

Make method to convert the Hex value to Long. (2)

Implement remaining Opcodes. (2)

Run main.py with filename argument from terminal. (4)

Update README file with proper documentation and instructions. (2)

Use more descriptive variable names. (2)

Final merge from develop branch into master. (1)

Print strings - HelloWorld!. (4)

Refactoring current code. (3)

Increase code coverage to 95% minimum. (3)

Codecov cleanup: create methods for repeat code. (2)

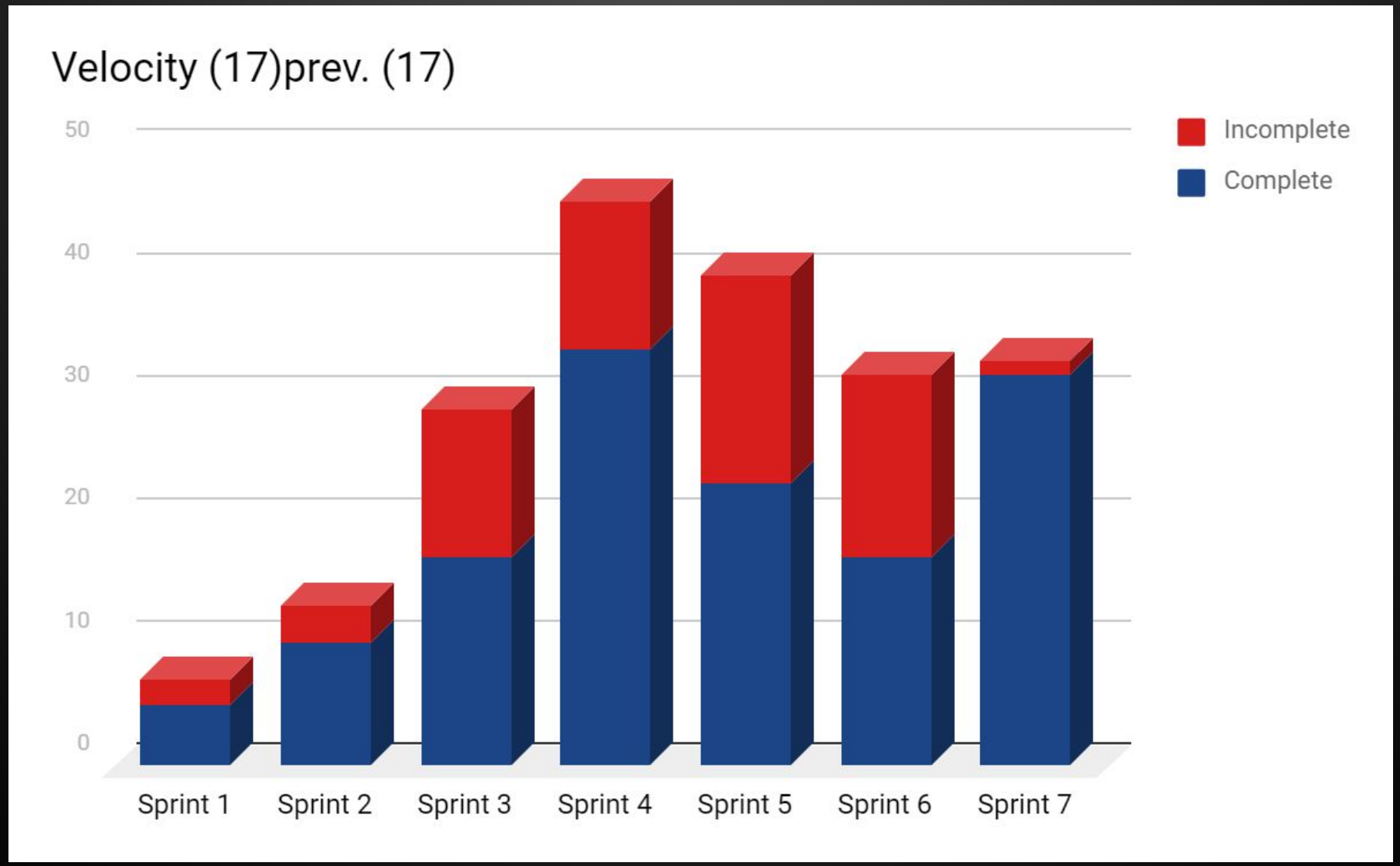
STORY POINTS

Incomplete:

Opcode getstatic (1)

DEMONSTRATION OF A RUNNING PROGRAM

SYSTEM DEVELOPMENT LIFE CYCLE STATISTICS


Velocity (17)prev. (17)
50
40
30
20
10
0
Incomplete
Complete
Sprint 1
Sprint 2
Sprint 3
Sprint 4
Sprint 5
Sprint 6
Sprint 7

SDLC STATISTICS

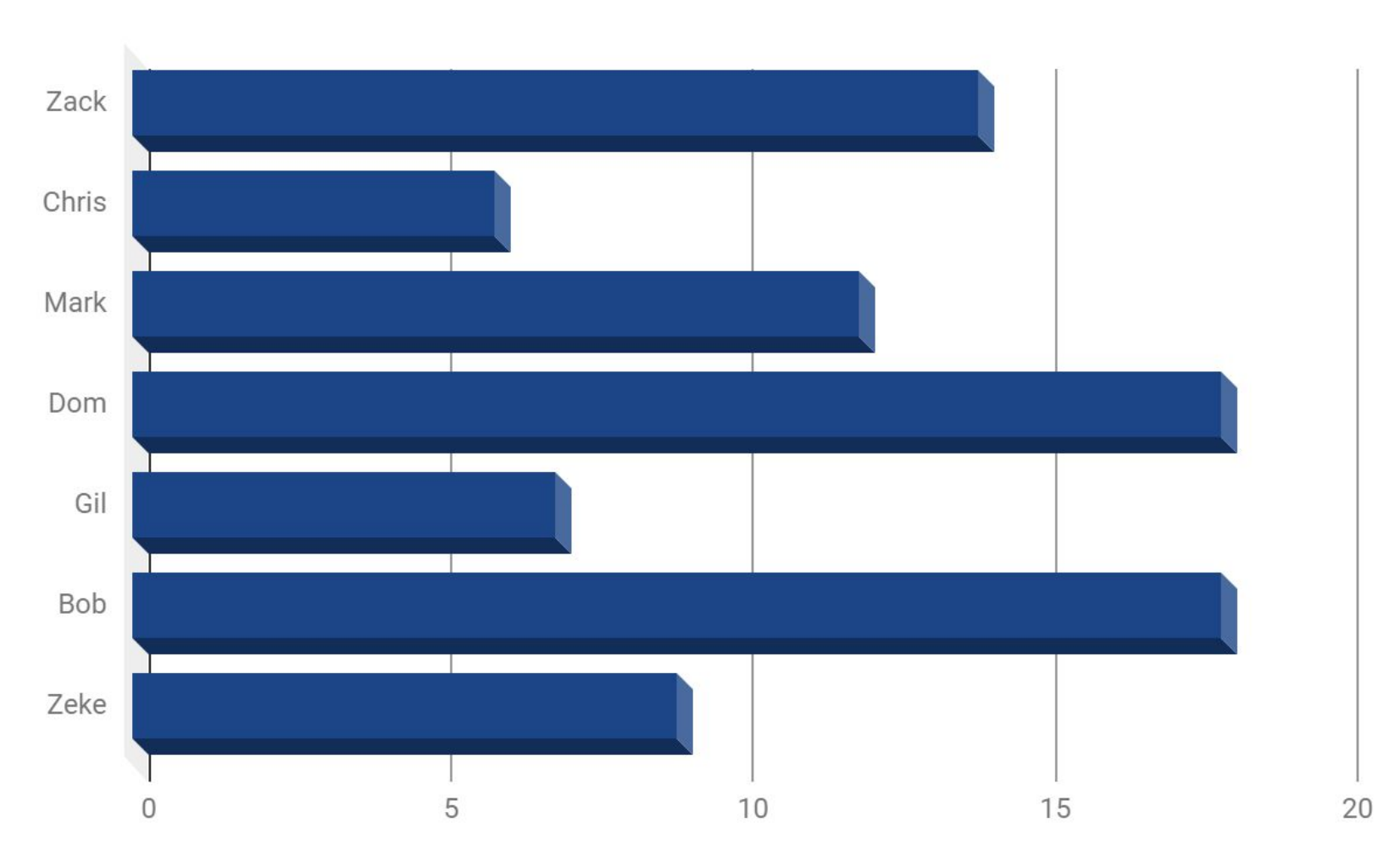
Sprint Backlog (0)

None

Project Backlog (1)

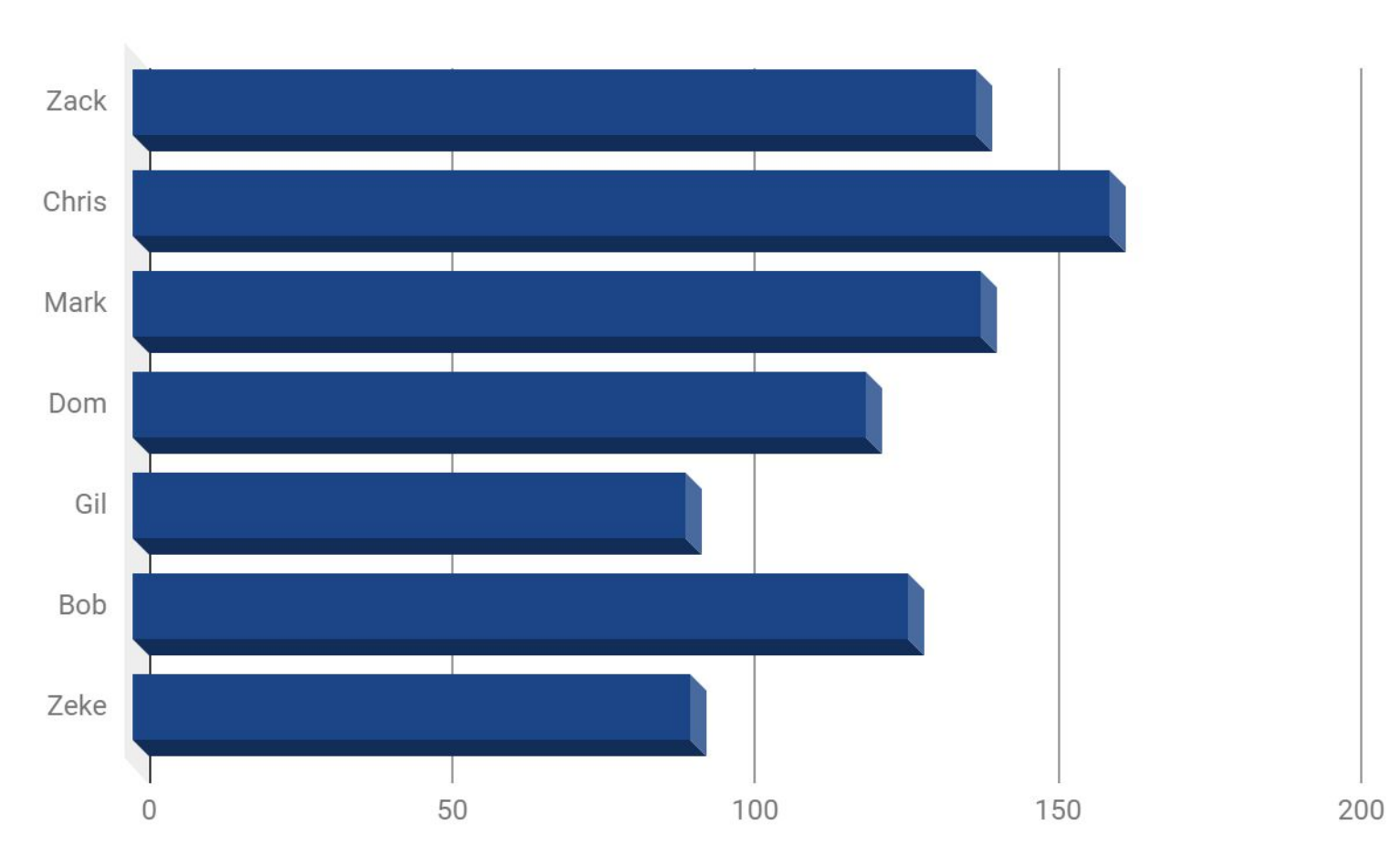
- Get static opcode

Team Hours



*additional 8 hrs each for 2hr Mon & Wed team meetings.

Team Course Hours

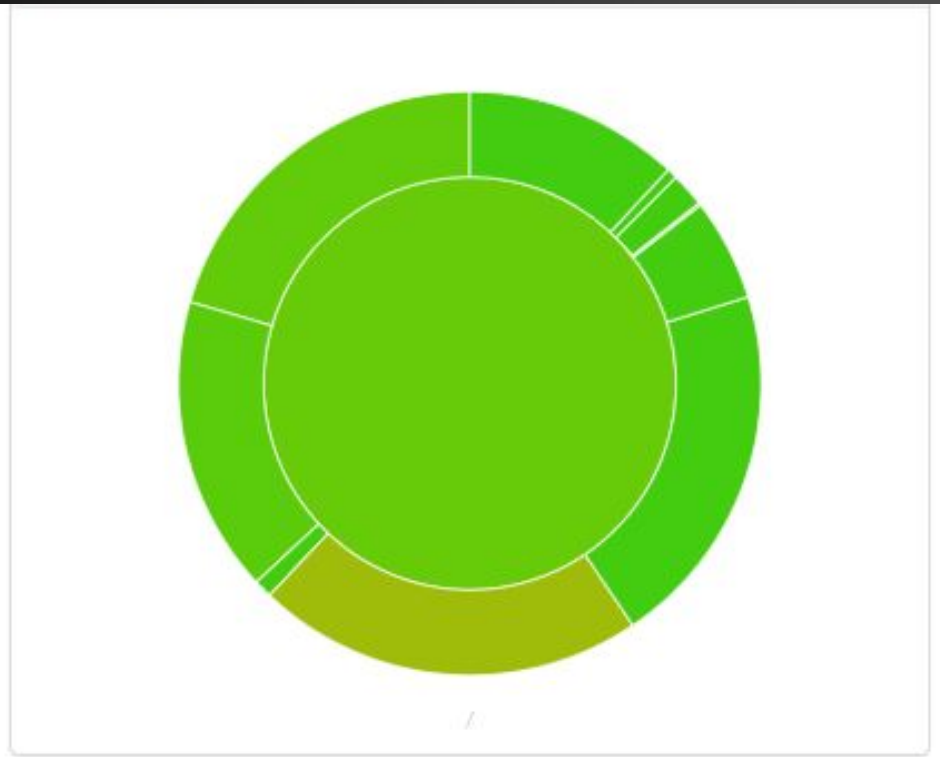


Coverage:

```
azuesflip@azuesflip-VirtualBox:~/cs3250/GitConfused$ coverage run --omit=*test*
-m unittest
..................................................
----------------------------------------------------------------------
Ran 50 tests in 0.412s

OK
azuesflip@azuesflip-VirtualBox:~/cs3250/GitConfused$
```

CodeCov (static)



Unable to find commit in GitHub — Browse Report
Unknown a day ago master c716072 CI Passed

messed up travis file, reverting. — Browse Report
zkhalido 2 days ago master 170dcab CI Passed

Merge pull request #28 from git-confused-team3/hotfix-duplicate-code — Browse Report
zkhalido 2 days ago master d792bfd CI Passed

fixing duplicate code from sonarcloud, new method to traverse method table — Browse Report
zkhalido 2 days ago master 6fcb532 CI Passed

Merge pull request #27 from git-confused-team3/develop — Browse Report
zkhalido 2 days ago master b50dbaf CI Passed

code smells; deleting commented code and unused variables. renaming opcodes to opcade on opcodes class. — Browse Report
zkhalido 2 days ago master b0fc46a CI Passed

View all recent commits

Files					Coverage
jvpm/packages	688	671	0	17	97.53%
Project Totals (10 files)	688	671	0	17	97.53%

Sprint Retrospective

WHAT WENT WELL

- Breaking down the user stories into story points
- Split workload more evenly between the Dev team.
- Communication between team.
- Overall team work.
- Completion of large amount of stories and backlog.

WHAT WENT POORLY

- Some testing didn’t run as planned on one test.

WHAT ARE WE GOING TO START/STOP/CONTINUE DOING

- Study for finals.
- Enjoy our Summer break!

Course Retrospective

WHAT WENT WELL

- Increase in communication
- New useful tools
- Better software practices
- Better team players
- Time management

WHAT WENT POORLY

- Testing
- In the beginning lack of communication

WHAT ARE WE GOING TO START/STOP/CONTINUE DOING

- Use the tools and methodologies to help us achieve our goals
- Writing less offensive code

Chris Johnson
5/7/19
CS 3250 -002

Hours Report and Retrospective
Team 3 Git Confused

Retrospective

What went well? -

- The most important thing that worked in our favor was having team meetings. We met on Mondays and Wednesdays before class for about two hours so that we could talk about the sprint objectives, break down the user stories and share what information we learned or any updates we had for the team. This proved to be the most important thing we could have done in my opinion. We were able to get everyone on the same page so that we would not have too many issues along the way.
- Another thing that went well was sharing information we learned. When someone learned something that could help the group we shared that information with each other. We also did pretty well when it came to sharing our problems or blocks in order to give someone else the chance to help or take over from where that person had left off.
- I think we had a fairly good group we all worked together well and did not have a lot of personal issues.

What went poorly? -

- In my opinion our biggest problem was not breaking down user stories enough to give each team member a manageable amount of work. We initially let people choose what they wanted to do without doing enough research to confidently say we could handle it.
- Another issue we had was with team meetings. Most of the team was able to meet at the same time but we did not put in enough effort to make sure everyone was updated or included.
- I think we did have some problems confronting each other with processes as we were a bit confused as to what to do. We did have conflicting ideas about backlog items specifically. We decided to work on new story points rather than making backlog items a priority which turned into a problem eventually but we later found out that backlog items should be a priority since the entire program would not work if we never completed the building blocks which makes far more sense.

What to Start/Stop/Continue Doing-

Start-

- For the future I want to do more research before beginning a project or user story. Most of my experience with CS2 was just trying then fixing things but I have learned that it would be far easier to research a topic before doing the work. This would also save a lot of coding time.
- Something else I want to start doing is tracking time more precisely. I was able to track most of my time but there were some times that I did not track because I did not feel it was quality work that I was putting in.
- I also want to keep better notes on the work that I did. There were some times that I did not give helpful notes in my commits because I thought I knew what I was doing.

Stop-

- Something I need to stop doing is being a too controlling. I was somewhat trying to work with everyone so I could help with any problems as well as make sure we were finishing what we needed to do but I think I may have been micromanaging the others a bit even though I was not the product owner. I was able to learn how all of our code works so I was able to help team members that fell behind but it was probably fairly annoying to them.
- I also need to stop writing code before the planning and research phase. I could have saved myself a lot of headaches and time if I had properly planned my ideas rather than just starting then running into problems

Continue-

- I think it was important to have a regular meeting time with the team in order to plan and share information so that is something I want to continue doing.
- Later in the semester I learned how we should be breaking up story points after we do research so I would like to keep that process in place. It will help each team member do an equal and minimal amount of work so that we stay efficient.

Toggle report

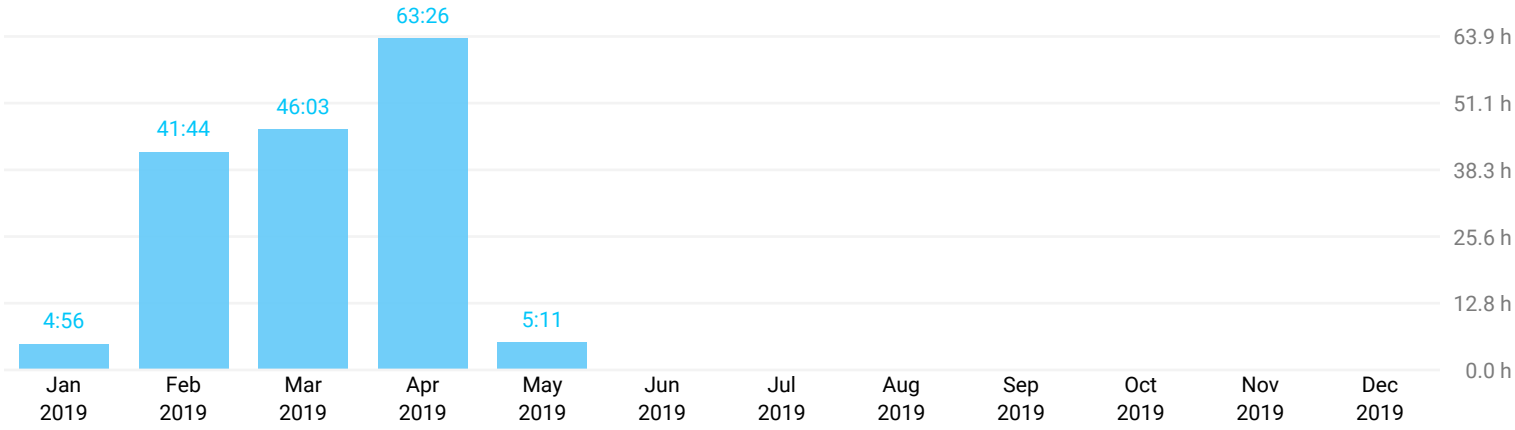
According to toggle I have about 160 hours for the group project but that report does not include class time. It also does not include time that I spent studying or reading the required materials.

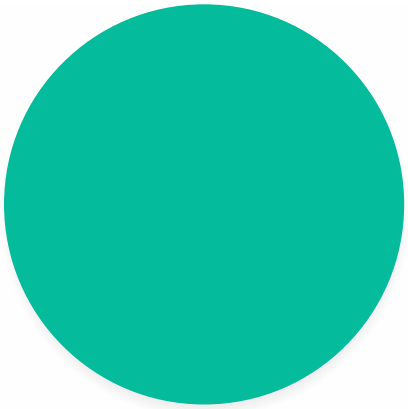
We also created a separate repo for our group that we put all of our notes and logs into but I went back and added my logs to toggle so that they would all be included in the toggle report below.

Summary Report

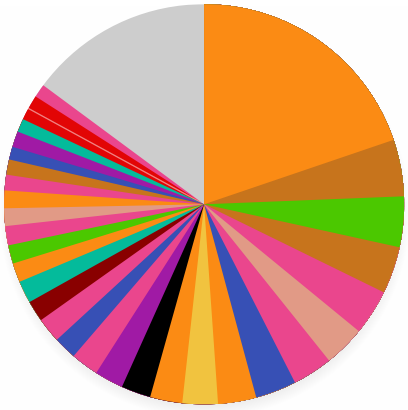
January 01, 2019 – December 31, 2019

TOTAL HOURS: 161:22:24





USER	DURATION
CJ Cjohn268	161:22:24



TIME ENTRY	DURATION
team meeting	31:50:41
working on getting to opcodes so we wont need to hard code them, fixed puller and getting pool size now, ended call with zack at 4:51 about an hour long just trying to analyze our code and see what we can do, new code working, fixing unittest 6:47, fixed all tests for new code now i just need to merge, finished for now just checking travis and cov	7:22:41
Adding unittest making sure everything is running for sprint 4	6:34:51
git confused(- made jvpm_methods a class that is instanciable - made a test for iadd to test instance - fixed dictionary to work with object - pushed to test branch - working on testing the method to get opcodes and run them * issue - still need to be able to read opcodes from class file - call with zack 2:15 - 2:46 -(bob should be done with opcode imprt tonight) - added test for loop to check if it is reading bytes correctly and going through dictionaries to return methods - also a lot of research for making objects of classes in python)	6:11:56
making sprint power point	6:01:12
making pool translator(-making an application to translate bytecode of constant pool -discord chat -ended call with bob and mark, mark is working on the rest of the opcode methods, i am working on translating the pool, finishing translator - got the translator to go through the dictionary and start getting next values - need to start building all the methods and append the dictionary with the new strings that are pulled)	5:21:55
more tests(- finished istore and load tets - had to fix loop test to not use fixed variables - fixed xor test issue, was just looking for wrong result - got travis to run tests on program by fixing imports and moving test.class up a folder - still have issues with sonar cloud not working from travis)	5:13:25

trying ot hook up sonar	5:11:45
reseaching mocking and unittest- figuring out everyones code to make tests -making tests in opcode branch -switched to header info to make tests -added stack class for hader info -ran into issues in general -made a xcel doc with breakdown of the bytecodes to excpect	5:07:49
working on finishing merge with bobs new reader, fixed merge issues and edited bobs code, 4:02 , just doing tests now to up coverage, merged to develop and coverage at 90	4:34:15
- starting to refactor cool translator - learned the difference between global variables and instance variables 3:39 - hit a block and had to draw out my problem	4:15:00
working on reading attribute tables, got first set of opcodes 2:40, having issues with line number table method	3:48:39
still working on decoding pool and translating ad two by hand to make sure	3:45:29
making tests in header info	3:34:00
meeting with zeek we are trying to run opcodes and prep for presentation	3:06:31
group meeting	3:04:21
fixed the pool pulling method and made tests, going back to see why sonar wont show coverage or does not see unittests	2:57:12
working on sprint 4 stuff and sorting constant pool	2:44:36
just playing with method table information, reader getting correct opcodes 5:49	2:32:34
call with zack made class that creates an object of opcode methods, (- call with zack - we made another branch - made jvpm_methods into a class than can create an object - made one test for iadd to test object and works)	2:30:34
working on translator	2:30:20
- continuing to refactor 11:30 - team meeting - 12:55 refactoring working but need rest of methods and getting hex bit in front of utf8 conversion - removed funky character in utf8 translation, so refactoring done need to build other methods 1:21 - 1:40 call ended - added field reference and seems to be working with add two	2:18:00
going to try and read other pools, got to method pool but ran out of time for today	2:06:49
trying to edit new pool reader to work in code, pool reader should be OK to add in	2:04:35
learning to branch/ more branch commands/ local branches / pull requests	1:57:04
merging everything to pull into master for luke	1:51:34
team meeting	1:51:33
team meeting, researching invoke virtual, doing some back seat coding	1:45:13
working with zack to pull bytes until 3, then working with gill to prep for presentation after 3(- making more unit tests but dont know how get pull works - continuing to build a new pool pulling method (just need to skip index for double and long) - new pool reader works, going to test it on old code)	1:45:00
fixing more unit test to work with refactored code	1:40:20
fixing test and prep for sprint	1:39:21
Other time entries	24:03:09

USER - TIME ENTRY	DURATION
CJ Cjohn268	161:22:24
- continuing to refactor 11:30 - team meeting - 12:55 refactoring working but need rest of methods and getting hex bit in front of utf8 conversion - removed funky character in utf8 translation, so refactoring done need to build other methods 1:21 - 1:40 call ended - added field reference and seems to be working with add two	2:18:00

USER - TIME ENTRY	DURATION
- starting to refactor cool translator - learned the difference between global variables and instance variables 3:39 - hit a block and had to draw out my problem	4:15:00
Adding unittest making sure everything is running for sprint 4	6:34:51
attempting mock	1:35:30
call with zack made class that creates an object of opcode methods, (- call with zack - we made another branch - made jvpm_methods into a class than can create an object - made one test for iadd to test object and works)	2:30:34
conference call	0:37:12
continuing to see why sonar wont show tests, call with gill 9:20 ended call with gill 10:24, preping presentation	1:35:36
debugging tests in doms repo	1:06:08
finishing sprint	0:15:13
finishing sprint/team meeting	1:34:00
fixed the pool pulling method and made tests, going back to see why sonar wont show coverage or does not see unittests	2:57:12
fixing more unit test to work with refactored code	1:40:20
fixing test and main issues	0:45:00
fixing test and prep for sprint	1:39:21
fixing unit test	0:52:57
git confused(- made jvpm_methods a class that is instanciable - made a test for iadd to test instance - fixed dictionary to work with object - pushed to test branch - working on testing the method to get opcodes and run them * issue - still need to be able to read opcodes from class file - call with zack 2:15 - 2:46 -(bob should be done with opcode imprt tonight) - added test for loop to check if it is reading bytes correctly and going through dictionaries to return methods - also a lot of research for making objects of classes in python)	6:11:56
going to try and read other pools, got to method pool but ran out of time for today	2:06:49

USER - TIME ENTRY	DURATION
got opcode runner to work	0:30:15
group meeting	3:04:21
installed sonar scanner	0:45:06
just playing with method table information, reader getting correct opcodes 5:49	2:32:34
learning to branch/ more branch commands/ local branches / pull requests	1:57:04
making jvpm methods into a class to test it	1:13:51
making pool translator(-making an application to translate bytecode of constant pool -discord chat -ended call with bob and mark, mark is working on the rest of the opcode methods, i am working on translating the pool, finishing translator - got the translator to go through the dictionary and start getting next values - need to start building all the methods and append the dictionary with the new strings that are pulled)	5:21:55
making sprint power point	6:01:12
making tests in header info	3:34:00
meeting with bob delegating work	1:00:51
meeting with zeek we are trying to run opcodes and prep for presentation	3:06:31
merge deleted some stuff trying to fix them again	0:35:44
merging everything to pull into master for luke	1:51:34
monday team meating	1:18:37
more tests(- finished istore and load tets - had to fix loop test to not use fixed variables - fixed xor test issue, was just looking for wrong result - got travis to run tests on program by fixing imports and moving test.class up a folder - still have issues with sonar cloud not working from travis)	5:13:25
reseaching mocking and unittest, figuring out everyones code to make tests	1:11:43

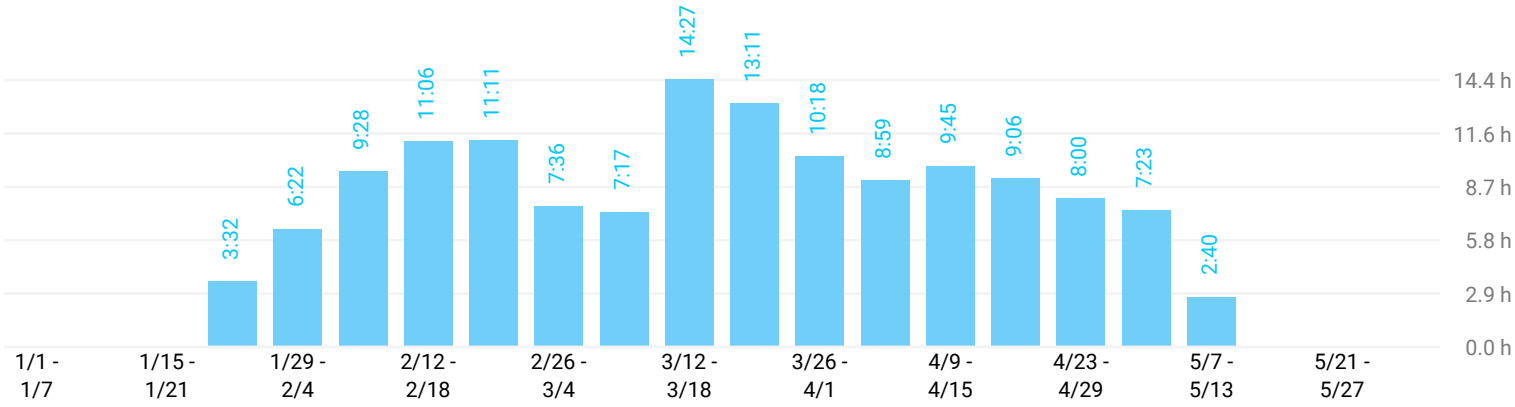
USER - TIME ENTRY	DURATION
reseaching mocking and unittest- figuring out everyones code to make tests -making tests in opcode branch -switched to header info to make tests -added stack class for hader info -ran into issues in general -made a xcel doc with breakdown of the bytecodes to excpect	5:07:49
reviewing bobs pool reader to see if it will work, but it uses a different reader than before and im not sure if the pool is indexed correctly	1:36:09
skype with gil anbd bob	0:36:09
still working on decoding pool and translating ad two by hand to make sure	3:45:29
team meeting	31:50:41
team meeting	1:51:33
team meeting, researching invoke virtual, doing some back seat coding	1:45:13
trying ot hook up sonar	5:11:45
trying to build more methods but I dont think im getting the correct constant pool for other class aside from add two and test	0:20:00
trying to edit new pool reader to work in code, pool reader should be OK to add in	2:04:35
trying to fix sonar cloud again...	1:25:08
trying to merge to develop, merge will be easy, we still need unit tests before the merge and we need to try and incorporate bobs code but it does not seem to be working for me	0:40:42
trying to run coverage of hello world program	0:45:51
working on code smells	1:03:43
working on decoding pool(-continuing to translate pool -issues decoding byte(as string) from utf-8 -decoding finally working for utf 8- had to convert to int then to hex then decode utf8), (- cont - 11:30 am- -got it to append a new dictionary with corresponding strings but am getting an extra string for some reason in slot 1)	0:40:55
working on editing the test class	0:28:42

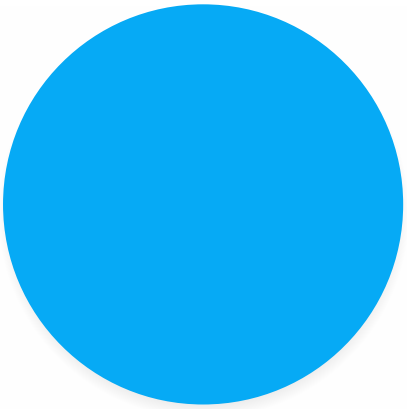
USER - TIME ENTRY	DURATION
working on finishing merge with bobs new reader, fixed merge issues and edited bobs code, 4:02 , just doing tests now to up coverage, merged to develop and coverage at 90	4:34:15
working on getting to opcodes so we wont need to hard code them, fixed puller and getting pool size now, ended call with zack at 4:51 about an hour long just trying to analyze our code and see what we can do, new code working, fixing unittest 6:47, fixed all tests for new code now i just need to merge, finished for now just checking travis and cov	7:22:41
working on reading attribute tables, got first set of opcodes 2:40, having issues with line number table method	3:48:39
working on sprint 4 stuff and sorting constant pool	2:44:36
working on translator	2:30:20
working with zack to pull bytes until 3, then working with gill to prep for presentation after 3(- making more unit tests but dont know how get pull works - continuing to build a new pool pulling method (just need to skip index for double and long) - new pool reader works, going to test it on old code)	1:45:00
Without description	1:28:07

Summary Report

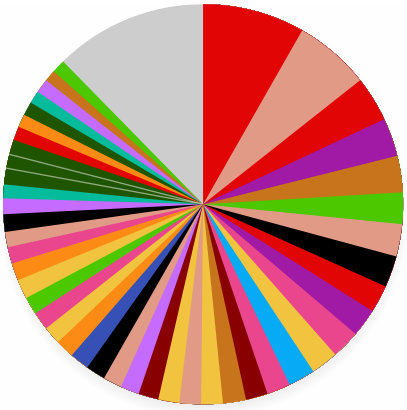
January 01, 2019 – May 31, 2019

TOTAL HOURS: 140:28:52





PROJECT	DURATION
CS 3250	140:28:52



TIME ENTRY	DURATION
Spike	11:35:00
Team meeting	8:29:43
Unit testing	5:11:56
Go through moodle to prepare for demo day	4:25:00
Opcode methods	4:07:16
parse constant pool	3:38:00
Discord meeting	3:37:41
Team Meeting	3:29:00
Code Cleanup	3:11:00
Implement stacks	3:11:00
opcodes	3:11:00
more overflow	3:08:50
Story points	3:03:00
Unit test edge case	2:36:00
Work on presentation	2:35:17
Work on powerpoint	2:34:55

Constant pool size	2:29:24
Unittest	2:19:34
Read the five dysfuntions of a team	2:18:20
Pool translate	2:18:14
Research struct	2:11:13
Learn about bytecode	2:11:00
Help finalize presentation	2:10:14
switch to vscode	2:09:05
Parse bytecode	2:08:15
Research pulling constant pool	2:05:56
Floating opcodes	2:03:00
More opcodes	2:03:00
spike	2:03:00
Group meeting	2:00:00
Overflow and bipush	1:51:46
Python functions for parsing constant pool	1:51:38
Discord questions	1:50:00
Remove unused lines	1:47:00
Research	1:45:00
Constant pool methods	1:44:32
Meeting with chris	1:37:00
Finalize presentation	1:35:37
python list function	1:34:20
More tag byte methods	1:32:27
Integrating Travis CI	1:30:51
Work with team on presentation	1:30:45
Reading on stacks	1:29:46
Start 2nd sprint	1:26:39
docstring to method	1:26:16
Other time entries	17:20:22

PROJECT - TIME ENTRY	DURATION
CS 3250	140:28:52
Checking email/downloading programs for CS	0:24:38
Clone github and edit python file	0:23:15
Clone repository through Ubuntu	0:08:14

PROJECT - TIME ENTRY	DURATION
Code Cleanup	3:11:00
Constant pool methods	1:44:32
Constant pool size	2:29:24
Convert bytecode to instructions	1:11:45
Discord meeting	3:37:41
Discord meeting with Chris and Bob to decipher code	1:20:06
Discord questions	1:50:00
docstring to method	1:26:16
Finalize presentation	1:35:37
Floating opcodes	2:03:00
git branch	1:13:28
Give up on ubuntu and wonder why I didn't start with BASH in windows 10	0:17:02
Go through moodle to prepare for demo day	4:25:00
Group meeting	2:00:00
Help finalize presentation	2:10:14
Implement stacks	3:11:00
Install pip/coverage	0:12:51

PROJECT - TIME ENTRY	DURATION
Install Ubuntu	0:08:33
Integrating Travis CI	1:30:51
Learn about bytecode	2:11:00
Meeting with chris	1:37:00
More opcodes	2:03:00
more overflow	3:08:50
More tag byte methods	1:32:27
Opcode methods	4:07:16
opcodes	3:11:00
Overflow and bipush	1:51:46
Parse bytecode	2:08:15
parse constant pool	3:38:00
Pool translate	2:18:14
Python functions for parsing constant pool	1:51:38
python list function	1:34:20
Read SCRUM guide	0:45:42
Read the five dysfuntions of a team	2:18:20

PROJECT - TIME ENTRY	DURATION
Reading on stacks	1:29:46
Refactoring	1:01:46
Remove unused lines	1:47:00
Research	1:45:00
Research pulling constant pool	2:05:56
Research struct	2:11:13
Retrospective	1:03:33
Reviewing team changes	0:24:11
Setting up Python to work in Command Prompt	0:13:24
Setting up Toggl	0:02:13
Sign up for Moodle/Github/Slack/Travis	0:12:09
Spike	11:35:00
spike	2:03:00
Start 2nd sprint	1:26:39
Start work on first github assignment	0:20:03
Story points	3:03:00
switch to vscode	2:09:05

PROJECT - TIME ENTRY	DURATION
Team meeting	8:29:43
Team Meeting	3:29:00
Team meeting (dynamic analysis)	1:18:12
Try to understand how Bob pulled constant pool	0:57:00
Unit test edge case	2:36:00
unit test for new methods	1:13:11
Unit testing	5:11:56
Unittest	2:19:34
Watch an intro to Github https://www.youtube.com/watch?v=0fKg7e37bQE	0:23:58
Work on powerpoint	2:34:55
Work on presentation	2:35:17
work on remaining opcode methods	1:22:42
Work on unit test	1:22:08
Work with team on presentation	1:30:45
Working with team for unittest	1:20:18