

# My CIT360 Semester Journal

---

By Justice Okai

# Wk1: SETUP



- Create a GitHub account.
- Make an availability post on the discussion board in order to be assigned to a group.
- Download and install NetBeans.
- Link up the NetBeans to my GitHub account and test.
- Create a google Hangout account: Remember Justice “Always attend to group meetings every week”



# Wk2: My Team

- We all agreed to communicating and working together using Google Hangout. We also agreed our meeting days will always be on Tuesdays and Fridays.

## Members

Alkire, Jeremy

Ankomah, Nana Baah

Gannon, Sophia

Hysi, Jetmir

Okai, Justice

Palmer, Robert

# Wk3: List of Code Topic discussed by My team on Tuesday, Two treated on same week. 'Friday'

---

- Java Collections
- Hibernate
- QCJSON & JSON
- HTTPURLConnection from Android
- JUnit Tests
- Threads, Executors, Runnables, and Updating the Android UI
- MVC
- Application Controller Pattern
- Servlets
- Use Case Diagrams
- Use Case Documents
- State Diagrams
- Sequence Diagrams
- System Level Tests

A	B	C	D	E	F	G	H	I
CIT360 Topics	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
Java Collections		Jeremy Alkire						
Threads, Executors, and Runnables						Jeremy Alkire		
Application Controller Pattern						Justice Okai		
MVC				Justice Okai				
JavaBeans		Robert Palmer						
Hibernate		Robert Palmer						
QCJSON and JSON							Nana Baah Ankomah	
Android HttpURLConnection								Spencer
Servlets				Spencer				
JUnit Tests					Nana Baah Ankomah			
System Level Tests					Huffman Kris			
Use Case Diagrams	Sophia Gannon							
Use Case Documents	Spencer							
State Diagrams							Huffman Kris	Hysi, Jetmir
Sequence Diagrams		Sophia Gannon						

## Wk4: Team Topic Schedules/Teaching Presentations

- We did our very best to uphold to this schedule and used a SCUM method to track, report and analyze our process for a successful plan.



## Wk5: My Individual Schedule

- This schedule I have planned and as well tried as much as possible to uphold. To help me gain good feedback from my wonderful instructor on how to improve and better my work.

A	B	C
CIT360 Topics	My Study Schedule	My turn in Schedule
Java Collections	Week: 3	Week:3
Threads, Executors, and Runnables	Week: 4	Week: 4
Application Controller Pattern	Week: 5	Week: 6
MVC	Week: 6	Week: 6
JavaBeans	Week: 7	Week: 7
Hibernate	Week: 8	Week: 8
QCJSON and JSON	Week: 8	Week: 8
Android HttpURLConnection	Week: 9	Week: 9
Servlets	Week: 3	Week: 3
JUnit Tests	Week: 9	Week: 9
System Level Tests	Week: 10	Week: 10
Use Case Diagrams	Week: 4	Week: 4
Use Case Documents	Week: 7	Week: 7
State Diagrams	Week: 6	Week: 7
Sequence Diagrams	Week: 5	Week: 5

# Wk6: Research topics reference /links

---

- *Collected links based on my research, inline with the topics.* My Team Teaching Presentation on MVC and Request for familiarity assessment.

Java Collections:

The official Collections tutorial\*(<http://java.sun.com/docs/books/tutorial/collections/index.html>)

Another Collections

tutorial\*([http://www.tutorialspoint.com/java/java\\_collections.htm](http://www.tutorialspoint.com/java/java_collections.htm))

Hibernate:

The Singleton

Pattern\*([http://en.wikipedia.org/wiki/Singleton\\_pattern](http://en.wikipedia.org/wiki/Singleton_pattern))

\*The Proxy Pattern\*([http://en.wikipedia.org/wiki/Proxy\\_pattern](http://en.wikipedia.org/wiki/Proxy_pattern))

The Factory

Pattern\*([http://en.wikipedia.org/wiki/Factory\\_method\\_pattern](http://en.wikipedia.org/wiki/Factory_method_pattern))

[\*HQL\*](<https://docs.jboss.org/hibernate/orm/3.3/reference/en-US/html/queryhql.html>)

definition\*([http://en.wikipedia.org/wiki/Many-to-many\\_%28data\\_model%29](http://en.wikipedia.org/wiki/Many-to-many_%28data_model%29))

[\*Many To Many example and

diagrams\*](<http://www.tomjewett.com/dbdesign/dbdesign.php?page=manymany.php>)

[\*Full Hibernate annotations declaration and description\*]([http://docs.jboss.org/ejb3/app-server/HibernateAnnotations/reference/en/pdf/hibernate\\_annotations.pdf](http://docs.jboss.org/ejb3/app-server/HibernateAnnotations/reference/en/pdf/hibernate_annotations.pdf))

# Wk7: Research topics reference /links continues

---

JSON Data Serialization: \*JSON Introduction (follow the links on the

left)(<http://www.w3schools.com/json/default.asp>)

HTTP Communication:

Simple Tutorial\*)(<http://www.vogella.com/articles/AndroidNetworking/article.html>)

Some Initial Options\*)(<http://android-developers.blogspot.com/2011/09/androids-http-clients.html>)

JUnit Definition (<http://en.wikipedia.org/wiki/JUnit>)

JUnit Home Page (<http://www.junit.org/>)

[\*Unit Testing\*)([http://en.wikipedia.org/wiki/Unit\\_testing](http://en.wikipedia.org/wiki/Unit_testing))

An example of a JUnit

test(<http://mars.byui.edu/barneyl/CaseSupports/StatisticsTest.java.zip>)

Safe Parallel Processing:

Threading example\*)(<http://java.sun.com/docs/books/tutorial/essential/concurrency/simple.html>)

Simple Runnable Example\*)(<http://www.roseindia.net/java/java-get-example/get-current-thread.shtml>)

Updating the view using Android

Handlers (<http://mindtherobot.com/blog/159/android-guts-intro-to-loopers-and-handlers/>) (Look

at the examples at the bottom of the page)



# Wk8: continues

## Teaching Presentation on ACP

---

Java(<http://java.sun.com/developer/technicalArticles/ALT/Reflection/index.html>)

The Model, View, Control (MVC) Software Design Pattern:

<http://www.bennadel.com/blog/2379-A-Better-Understanding-Of-MVC-Model-View-Controller-Thanks-To-Steven-Neiland.htm>)

Socket IO: [\*Serialization\*]([http://en.wikipedia.org/wiki/Serialization#Example\\_2](http://en.wikipedia.org/wiki/Serialization#Example_2))

UML:

UML([http://en.wikipedia.org/wiki/Unified\\_Modeling\\_Language](http://en.wikipedia.org/wiki/Unified_Modeling_Language))

UML Class Diagrams\*]([http://en.wikipedia.org/wiki/Class\\_diagram](http://en.wikipedia.org/wiki/Class_diagram))

UML Sequence Diagrams\*]([http://en.wikipedia.org/wiki/Sequence\\_diagram](http://en.wikipedia.org/wiki/Sequence_diagram))

UML State

Diagrams\*]([http://en.wikipedia.org/wiki/State\\_diagram#UML\\_State\\_diagram](http://en.wikipedia.org/wiki/State_diagram#UML_State_diagram))

UML Use Case Diagrams\*]([http://en.wikipedia.org/wiki/Use\\_case\\_diagram](http://en.wikipedia.org/wiki/Use_case_diagram))

# Wk9: Activities

---

- Working on Code Topics and UML Diagrams based on wonderful instructor's feedback as well as working on an individual APP.





# Wk10: Justice 'Make submission' and Reach out to Instructor

---

- Fluency Review
- Professional Behavior
- Personal Plan
- Written Proposal.

Wow!!! Achieved a basic level in all topics, hope to progress more...



**WOW!**

# Wk11: Working on APP

---

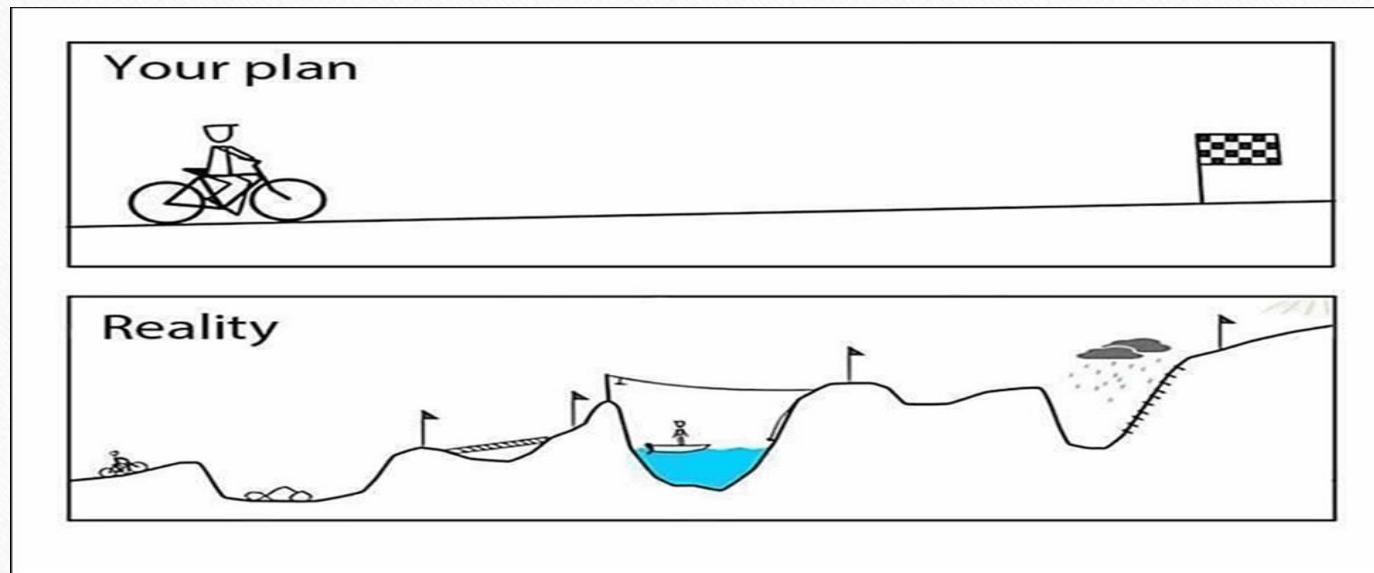
- Dragon Knight Application.

Simple mobile adventure game.



# Wk 12: Weight

- The end of the semester on the edge and getting everything done and checked to my portfolio.



## Wk. 13&14

---

- Without the help of team as well as my wonderful instructor, I wouldn't have had the opportunity to make it. Am so grateful to my team as well as my wonderful instructor with his inspiring feedback.

**I MADE IT**