

Multiplayer Android Game

G. Dhanushka Lahiru

Department of Computer Science and Engineering, Moratuwa, Sri Lanka
dhanushkalahiru.12@cse.mrt.ac.lk

Abstract—Games with cards are a good entertainment method that works for anywhere. There is a variety of card games that can be played by different number of people. With the growth of smart phone technology people tend to use mobile apps to play games. The objective of the project is to develop a playful Multiplayer Android gaming app offered free to users on Android market. Game is based on the famous “Omi Game” which has already developed as a single player game. To give users a better game experience this project has delivered multiplayer version of the game where actual human players can play against each other. In order to achieve the expected deliverables, Java is used as the programming language and Android Studio is used as the IDE for the Android platform. To connect and pass messages between devices WIFI technology is used with Client-Server architecture. Output of the project has delivered ability to play famous “Omi Game” without having card pack in pocket, only four players who have android smart phones are enough.

Keywords—Android, Multiplayer, Omi, Java, Android Studio, WIFI, Client-Server, Smart Phone, Wifi Tethering

I. INTRODUCTION

Games with cards are a good entertainment method that works for anywhere. There is variety of card games that can be played by different number of people. With the growth of smart phone technology people tend to use mobile apps to play games.

“Omi” is a famous card game among Sri Lankans. To play it a card pack is necessary. Existing android games based on this game are single player games which will not give the players the real experience of the game due to Artificial Intelligent (AI) players. In many cases the movements of the AI player are predictable and thus reduce entertainment. Playing a multiplayer game is always fun. Instead of beating AI-controlled opponents, the player must face strategies created by another human being. Thus the future for multiplayer gaming looks bright indeed and multiplayer game fans are so enthralled with it that they often have no interest in single player games. Playing it anywhere anytime is more fun. So to have a famous card game in your smart phone which is usually cannot be played without a real card pack is going to be really amazing. Moreover with the rapidly developing technology it is more possible that the new world trends will tend towards the touch technology.

This paper introduces the multiplayer version of the android game “Omi” which will be available free at Google App Store. Single player version of the game already exists in the Google App Store [1, 2]. But the real gaming experience can only be achieved with the multiplayer version. Another main advantage is whenever a card pack is not available this can be used as an alternative if players have android phones with them.

Paper structure is as follows. Section II describes the related literature of the game, section III describes the system requirements and system design, section IV explains the system implementation, Section V summarizes the testing and analysis details of system, Section VI is concludes the paper.

II. LITERATURE REVIEW

A wide variety of related work is available in literature on android games [1-9]. Among them the authors in [1, 2, 6 and 8] have shown a system that facilities single player card games. Another interesting work has done in [3, 4, and 5] that has multiplayer capability of games. Snake Multiplayer [3] is a game which enables two players to play through WIFI or Bluetooth which makes gaming really fun and it has no cost other than like [4, 5, 7] which cost us they need internet connection.

There are many different types of games available in Android market such as Mino battle [4] which is a block game, Sky cue club: Pool and Snooker [5] which is a billiard game. Among them there are popular card games such as 6 Takes [6] and Solitaire Mega Pack [8]. However, some of them are not freely available.

The existing “Omi Game” is a single player game which is not really has much attraction. Because actual fun of Omi Game can only be get by playing with real players rather than a machine player.

This “Omi Game” set of rules [6] which can be learned through playing several times. It is a game of four players where they compete with each other as two teams each consisting of two players.

Although Bluetooth technology can be used to connect devices, not many smart phones have the ability to connect multiple devices at the same time over Bluetooth. Most of the smart phones allow only to connect two devices through Bluetooth at one time. So Bluetooth is widely used in two player games [10-12]. As “Omi” is a four player game it is better to used Wi-Fi technology which supports to connect four smart phone devices easily.

III. SYSTEM MODELS

A. System Requirements

Functional requirements of the system can be described in two parts. As project is adhered to the Client-Server model, there is a server and three clients. The stand-alone server aspect of the project is of the highest priority. The server must be able to run before clients will be able to connect to the server. The server will receive and send messages as events occur in the system.

Main functional requirements of server device are responding to client requests, forwarding events from clients appropriately, handling joins/quits appropriately and logging all important messages to its log area.

The clients, when initially launched application, will get a message saying “Server Connected” if they get connected to the server then they can send a nickname which they will be known as. When the user clicks the “Start” button, he will get a message of whether he joined or not to the game. Main functional requirements of client devices are allowing the user to send a nickname, updating the game area properly to show game status with other players’ moves and allowing the user to click on a card to put it on game table one at a turn when his turn came.

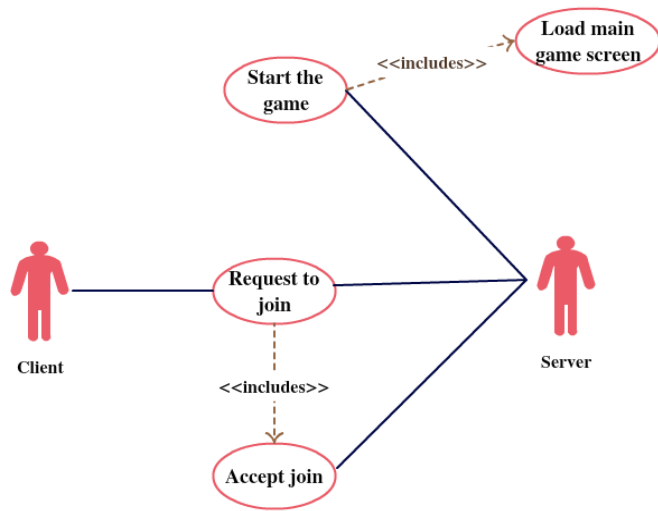


Fig. 1. Usecase diagram of the system

After clients chose “Join Game” option from starting menu they will request to join then server will decide to accept or not. Only the first three client requests will be accepted and after accepting 3 requests server will start the main game screen. As shown in Fig. 1, the main usecases are Request to join, Accept join, Start game and Load main game screen.

Non-functional requirements of the system are players should not leave the Wi-Fi zone before the game end, players need to have good experience in Omi game and players should make moves as soon as possible when their turn come unless others have to wait for him/her more time. “Omi” is an android mobile application and a multiplayer game. Since this system does not record any user activities and privacy data, it does not affect the privacy issues of the user.

B. System Design

This application uses three-tier architecture for its design. The architecture has three layers, a user interface layer (view layer), a business logic layer, and a data storage layer. Use case design, Logical view, and process views can be used to represent the application communications in the three layers. The layers in the architecture diagram are independent and uses well defined interface for inter communication. The layering makes it easy to achieve important features such as portability and reusability of the system. Fig. 2 shows the class diagram of the system, which

comprises of Player, Card, Team, Game and MainActivity classes.

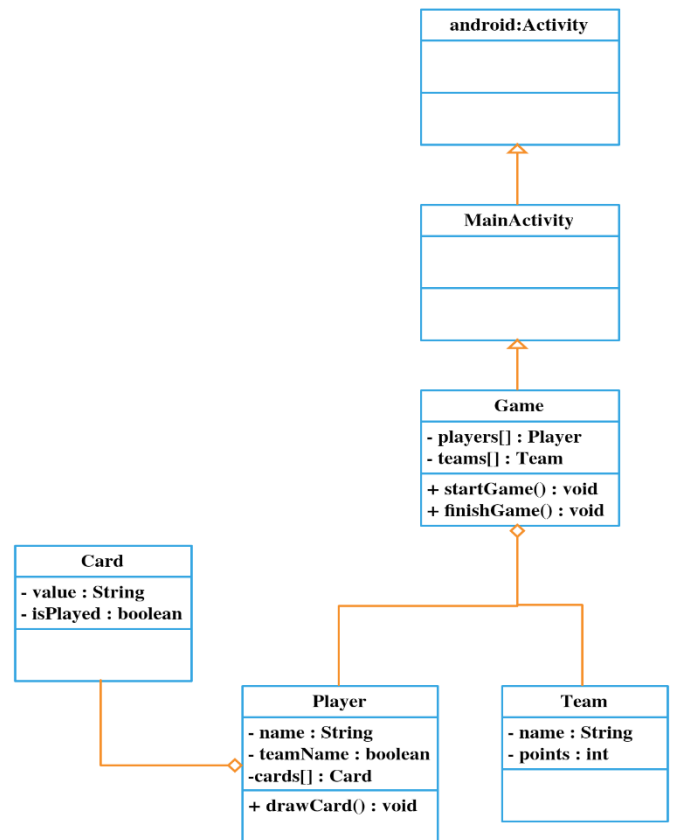


Fig. 2. Class Diagram (Logical View) of the system

Fig. 3, shows the main activity diagram of the system. The main activity starts the game. At the start, game area will be set respective to the players’ positions such that respective partner is in the front. In the initializing phase a card pack will be initiated and cards will be distributed among players. Players will draw cards when their turn come and this process will continue till one team score 10 points.

First a player has to select whether he is a server or client, if he is a server he has to start server and accept clients join requests who have start game as clients. After three clients have joined server player will start game and main game screen will be loaded in both server and client devices. Then they play the game till one team score 10 points then after that game will be finished

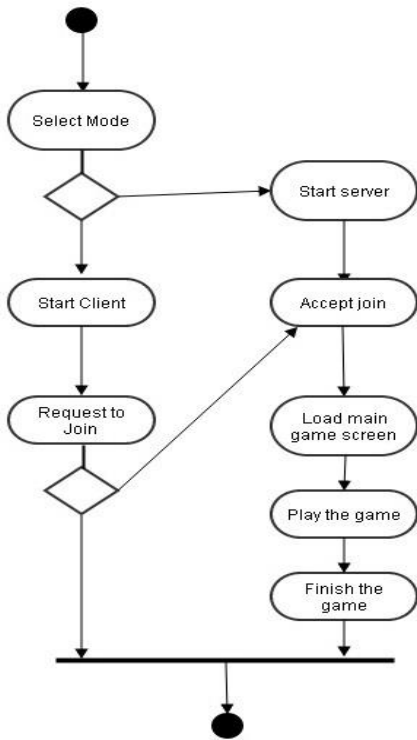


Fig. 3. Main activity diagram of the system

IV. SYSTEM IMPLEMENTATION

A. Implementation Procedure

The system is implemented using Java as the programming language and Android Studio as the IDE for the Android platform. Initially, Android Virtual Device (AVD) to test every step of the system. However, later on actual Android devices with USB Debugging feature is used to optimize the memory usage. Android SDK version 22 is used as the compiled version and the target SDK version with enabling system work on devices which have Android version 2.3 or above. Android version 2.3 API 9 (Gingerbread) is selected as the minimum SDK as it is the major version of Android in the market. Almost all of the devices which are in use, have this version or an above one. Therefore, all the android devices in use can play this game without any problem.

The application framework is designed using common libraries of android and the networking part is designed using Java Server Socket and Socket classes. The incoming and outgoing messages are handled using Vector class as it is Thread Safe. The implementation is incorporated with concurrency that enables multiple process to run in parallel and synchronized methods to provide thread safety. Singleton design pattern is used as only one Card Pack object is needed for a game.

When developing the “Omi Game” the images of classic cards were downloaded [13] as png images of size 72 x 96. Triumph card images were found from different sources and they were in different sizes [14-17]. All images were resized to match with game needs.

```

WHILE (Team A or B has not achieve 10 points)
  Create a new game round
  Shuffle Card pack
  IF (roundNumber % 4 = 0) THEN
    Set Triumph Selecting Player as 4
  ELSE
    Set Triumph Selecting Player as (round
    Number % 4)
  END IF
  Set player cards and send message
  Wait till game changes occur in clients
  IF (triumph selecting player is current
  player) THEN
    Select Triumph
    Wait till selection is send to others
    Set next turn of this sub round
  ELSE
    Triumph selecting chance is to one of
    client
    Set host players cards in hand
    Wait while selected triumph is passed to
    others
  END IF
  //Round completes when 8 sub rounds finished
  WHILE (Sub round count is not 8)
    //Sub round has 4 turns
    WHILE (turn < 4)
      Make a move
      INCREMENT turn
      Wait while a selection is done by
      respective player
      Message it to others
      //Set next turn in a sub round
      IF (round turn is 4) THEN
        Set round turn equal to 1
      ELSE
        Set round turn=current round Turn+1
      END IF
    END WHILE
    //End of a sub round
    //End of a sub round
    //Default value for nextTurn
    Set nextTurn = 1
  //Determine the winner of sub round and set hands
  FOR (j = 2; j <= 4; j++)
    IF (Value of the j player's card > nextTurn's
    value) THEN
      nextTurn = j;
    END IF
  END FOR
  Set next sub round starting player
  Set turn = 0
  After a sub round set players hands count and send
  message to others
  Reset the game table after a sub round
  Send message to others to reset
  INCREMENT Sub Round Number
END WHILE
//End of eight sub rounds
Set Winning team of a round and set score
Keep record when draw
Reset the game table after a complete round
INCREMENT roundNumber
END WHILE
//End of game
Identify winner team, send message, end game
  
```

The main algorithms of the system are Create connection and Game logic. In Create connection, the server creates a server socket and in clients create socket using Socket class using server's IP. By using accept method of server socket class server device establish a connection between server and client. The pseudocode for the Game logic is shown in Fig. 4.

B. Main Interfaces

This section describes some of the main interfaces as shown in Fig. 5 – Fig. 13.



Fig. 5. Main Menu



Fig. 6. Host Game



Fig. 7. Join Game



Fig. 8. Help

Fig. 5 shows the main menu of the game. It consists of three choices Host Game, Join Game or Help. Help contains instructions on how to play the game. The player who initiates the Wi-Fi hot spot has to start game by selecting Host Game while others who connected to the Wi-Fi hot spot have to select Join Game option.



Fig. 9. Server Connected



Fig. 10. Game Screen



Fig. 11. Sub Round



Fig. 12. Move

Fig. 10 shows the game screen. It consists of several elements. In the left upper corner there is the score board of the game. In right upper corner there is the triumph for the current game round. In the center of the screen there is the game table where the played cards are displayed with respect to each player and in the bottom there is the cards which the player is holding. Names of the other players are shown with respect to the current player's position and it helps to understand whose turn is next. Under names there are the number of hands won by each player in a round.

V. SYSTEM TESTING AND ANALYSING

Unit testing has done in the implementation process while developing the code and it was used to test functionalities module or class individually at completion of unit by the developer. The primary goal of unit testing is that taking the smallest piece of testable software in the application, isolating it from the rest of the code and determining whether that piece of code performs as expected [18-19].

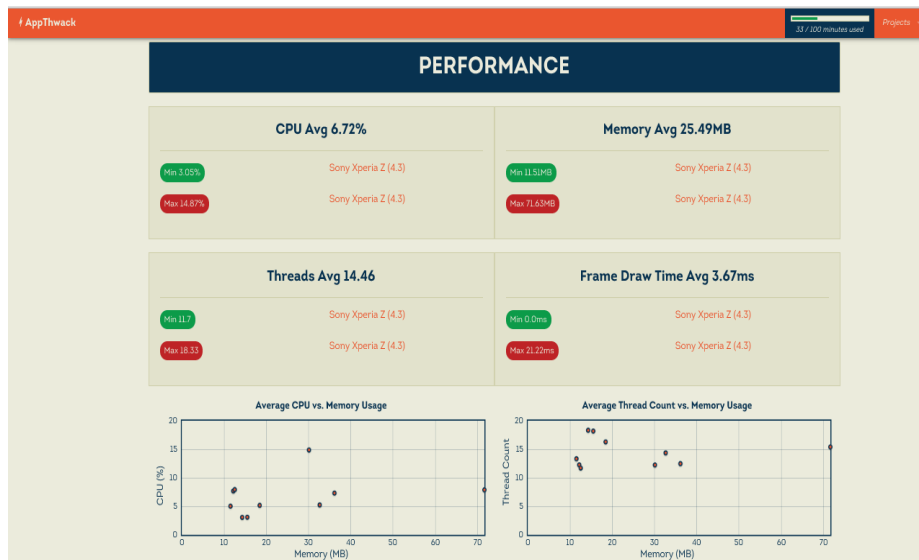


Fig. 14. Performance Testing

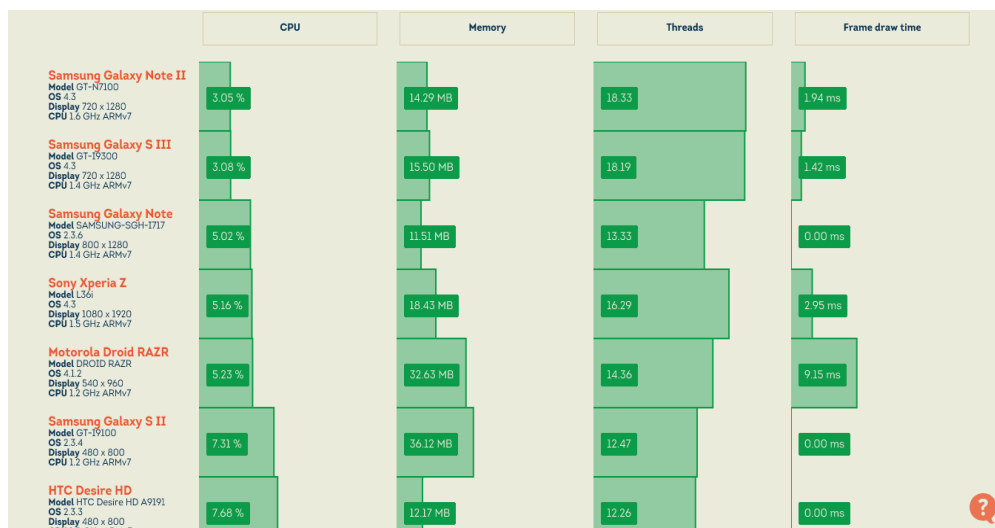


Fig. 15. Devices Based Testing

Objective of carrying out testing is to verify the code and the overall software works properly. To verify operational accuracy unit testing was carried out. Unit testing will be applicable if the code is implemented lowly coupled. (Lesser dependencies). So the system architecture was designed in a way that unit testing can be easily adopted. Android Studio by default contains JUnit testing library and the IDE is configured with proper settings to run JUnit unit testing.

Automated testing tool Appthwack.com is used for performance testing. It provides different types of automated tests which are run on devices in the cloud as shown in Fig. 14 and Fig. 15.

Although this is Wi-Fi based multiplayer game, it is well secured with network issues. Nowadays, Smart phones come with Wi-Fi tethering functionality which can be used in creating Wi-Fi hot spot [20]. This Wi-Fi hot spot can be shared with multiple devices. Only allowed members can connect to the

hosted hot spot by one player and able to play the game. As Wi-Fi hot spot can be secured with WPA2 PSK no security issues arise. Only the players who know the password can connect to the hot spot.

If the game hosting player suddenly ends game or disconnect the Wi-Fi hot spot, the game will be automatically ended. It is a failure due to player's actions and no other failures were detected.

VI. CONCLUSION AND FUTURE WORK

This paper describes the design and implementation details of a multiplayer android game known as Omi which is a card game of four players. This system mainly support with functionalities such as play together with four players as two teams each consisting two players connecting over Wi-Fi connection through smart phones. This application is developed using Java and Wi-Fi technology. This delivers a good solution

for the need of multiplayer game which already has been developed as a single player game [1, 2].

The Omi multiplayer game can be extended in many ways. The GUI of the game application can be improved by adding animations and the entertainment of the players can be increased by adding music clips to the application. The usability of the application can be improved by allowing less than four real players are there and filling the missing players virtually using an AI agent. A reliable communication protocol over the current IP stack can be implemented as a solution for the poor network links which might hinder the gaming experience. The game can be further improved by utilizing Bluetooth in the absence of WiFi allowing playing the game in public places or outdoor areas when the players are located together. Further, game rules can be improved to avoid cheating while playing.

REFERENCES

- [1] Sri Droiders, "Omi, The card game in Sinhala " Omi, The card game in Sinhala - *Android Apps on Google Play*, 09-July-2014. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.game.omisinhala>. [Accessed: 20-Feb-2016]
- [2] R. Adeef, "Omi Card Game ," *Omi Card Game - Android Apps on Google Play*, 07-Oct-2012. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.adeef.game>. [Accessed: 20-Feb-2016]
- [3] P. Oriol, "Snake Multiplayer - Android Apps on Google Play," *Snake Multiplayer - Android Apps on Google Play*, 16-Dec-2013. [Online]. Available at: <https://play.google.com/store/apps/details?id=mygames.snake>. [Accessed: 20-Feb-2016]
- [4] "Mino battle download free," *Mino battle Android apk game. Mino battle free download for tablet and phone.*, 10-Mar-2015. [Online]. Available at: http://play.mob.org/game/mino_battle.html. [Accessed: 20-Feb-2016]
- [5] "Mino battle download free," *Mino battle Android apk game. Mino battle free download for tablet and phone.*, 10-Mar-2015. [Online]. Available at: http://play.mob.org/game/mino_battle.html. [Accessed: 20-Feb-2016]
- [6] A. S. Heidelberg, "6 takes! - Android Apps on Google Play," *6 takes! - Android Apps on Google Play*, 26-Jan-2015. [Online]. Available at: https://play.google.com/store/apps/details?id=de.application_systems.six_takes. [Accessed: 20-Feb-2016]
- [7] AxesInMotion Casual, "Multiplayer Driving Simulator - Android Apps on Google Play," *Multiplayer Driving Simulator - Android Apps on Google Play*, 05-Nov-2015. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.aim.multiplayer>. [Accessed: 20-Feb-2016]
- [8] Tesseract Mobile, "Solitaire MegaPack - Android Apps on Google Play," *Solitaire MegaPack - Android Apps on Google Play*, 17-Feb-2016. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.tesseractmobile.solitairemulti>. [Accessed: 20-Feb-2016]
- [9] Zechner, Mario, and Robert Green. *Beginning Android Games*. Apress, 2011.
- [10] Haksoft, "Bluetooth chess - Android Apps on Google Play," *Bluetooth chess - Android Apps on Google Play*, 10-Jan-2016. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.hak.bluetoothchess>. [Accessed: 20-Feb-2016]
- [11] AppsOn, "Bobby Volleyball - Android Apps on Google Play," *Bobby Volleyball - Android Apps on Google Play*, 20-Sep-2014. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.appson.bobbyvolley>. [Accessed: 20-Feb-2016]
- [12] Ilumnis, "Bluetooth Tic Tac Toe Free - Android Apps on Google Play," *Bluetooth Tic Tac Toe Free - Android Apps on Google Play*, 07-Dec-2012. [Online]. Available at: <https://play.google.com/store/apps/details?id=com.ilumnis.xofree>. [Accessed: 20-Feb-2016]
- [13] "Classic Cards," *Classic Cards*. [Online]. Available at: <http://www.jfritz.com/cards/>. [Accessed: 27-Jul-2015]
- [14] "card symbol club glossy," *card symbol club glossy* [Online]. Available at: http://www.wpclipart.com/recreation/games/card_icons/card_symbol_club_glossy.png.html. [Accessed: 17-Sep-2015]
- [15] "Stock Vector By Miskokordic," *Bigstock*. [Online]. Available at: <http://www.bigstockphoto.com/image-14757635/stock-vector-poker-card-sign-spade>. [Accessed: 20-Feb-2016]
- [16] "Stock Images - golden crown," *Stock Images - golden crown*. [Online]. Available at: <http://www.shutterstock.com/similar-118790041/stock-vector-golden-crown-vector.html>. [Accessed: 20-Feb-2016]
- [17] "red-heart," *CBHS Best Buddies*, Mar-2013. [Online]. Available at: <http://cbhsbestbuddies.com/home/red-heart/>. [Accessed: 20-Feb-2016]
- [18] "Unit Testing," *Unit Testing*. [Online]. Available at: [https://msdn.microsoft.com/en-us/library/aa292197\(v=vs.71\).aspx](https://msdn.microsoft.com/en-us/library/aa292197(v=vs.71).aspx). [Accessed: 20-Feb-2016]
- [19] "Unit Tests," *Unit Tests*. [Online]. Available at: <http://www.extremeprogramming.org/rules/unittests.html>. [Accessed: 20-Feb-2016]
- [20] B. Nadel, "Wi-Fi tethering 101: Use a smartphone as a mobile hotspot," *Computerworld*, 04-Nov-2011. [Online]. Available at: <http://www.computerworld.com/article/2499772/mobile-wireless/mobile-wireless-wi-fi-tethering-101-use-a-smartphone-as-a-mobile-hotspot.html>. [Accessed: 20-Feb-2016]