

4 responses

[View all responses](#)[Publish analytics](#)

Summary

Your Name:

Nate Foss

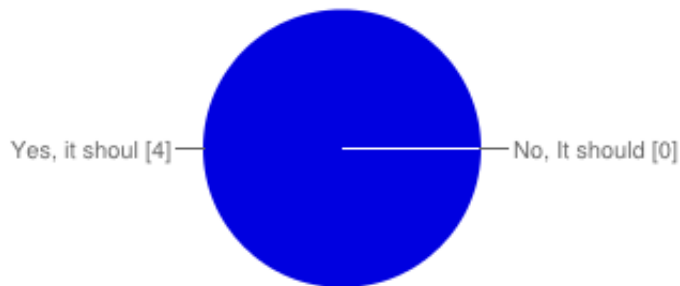
James Collins

Logan Sims

Sam Dixon

Mackenzie Revoyr

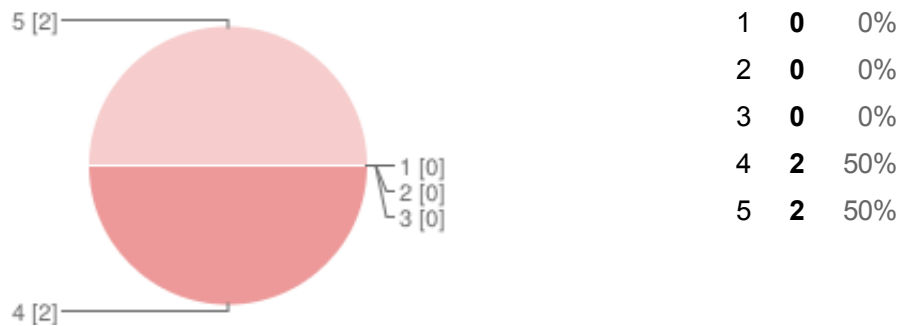
Should the assignment receive credit?

Yes, it should receive credit. **4** 100%No, It should not receive credit **0** 0%

Awesomeness

Yes!?!?! **3** 75%

On a scale from 1 to 5 please rate the look and feel of the app



Praise for submitter

Mackenzie did a fantastic job and deserves the awesomeness award! The level of effort she put into this is great! The remarks she gives to users are quite funny as well! I liked the pictures as well as they are very different than your traditional hangman

Did a great job of taking a simple hangman app and making it their own. The difficulty setting and categories is a very nice touch. The custom hangman image helped add to the charm of the app as well. Dictionaries.java was a nice way of managing the words and categories. All around very nice app.

I really like that you took time to make the game more interesting than the typical hangman game.

The game looks good and functions well. All the game options you have are cool.

Suggestions and Criticism

Keep up the good work!

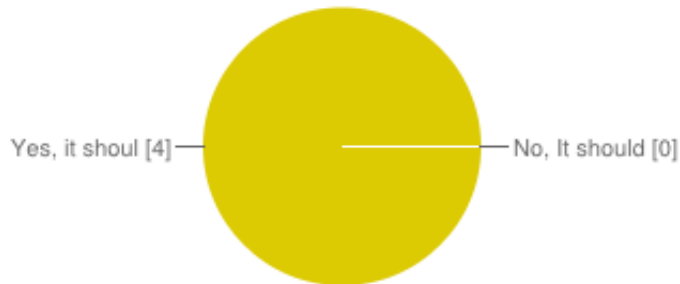
Since I've never seen those pictures before I wasn't sure how many lives I had left. I would have liked to have had a number somewhere that told me how many lives I had left before I killed the hangman.

When the key board comes up it covers some of your buttons and the key board can't be closed. Also when i change category while in a game it doesn't appear to change words, but that's just a preference of mine.

One suggestion I have would be to add an on-click listener to the Guess EditText. That way the user could just hit enter and not have to keep clicking the GUESS button. I think it's as simple as adding: `android:onClick="Guess"` to the Guess EditText in the xml file. Another thing I found is that once the game has ended you can keep guessing. Meaning after I have lost I could keep going until I guessed the word and the game would say I won. I would suggest disabling the Guess EditText once the game is over. This can be done with something like: `EditText t = (EditText)findViewById(R.id.guess); t.setEnabled(false);`

Amerie Lommen

Should the assignment receive credit?

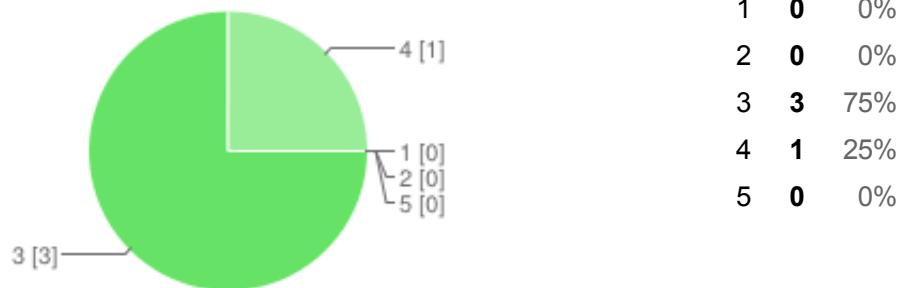


Yes, it should receive credit.	4	100%
No, It should not receive credit	0	0%

Awesomeness

No responses for this question.

On a scale from 1 to 5 please rate the look and feel of the app



Praise for submitter

The app's layout makes it look clean and easy to use. The code is clean and broken up into methods in a logical way. Nice use of the Boolean gameOver to stop the processing of new guesses.

Very quick and responsive. I like that you choose to not do a toast to display messages as this can add some headaches if the user guesses quickly and as such could cause multiple toasts to occur.

i really enjoy that you display the word after i fail.

The app works smoothly, and the layout is easy to navigate.

Suggestions and Criticism

If I input multiple letters (which is allowable), none of them are checked for correctness, but will fail. For instance, if the word is 'CAT', and I input 'AAA', a hangman piece will be added. But if I simply enter 'A', it will succeed. Either disallow multiple letter input, or change the method of input.

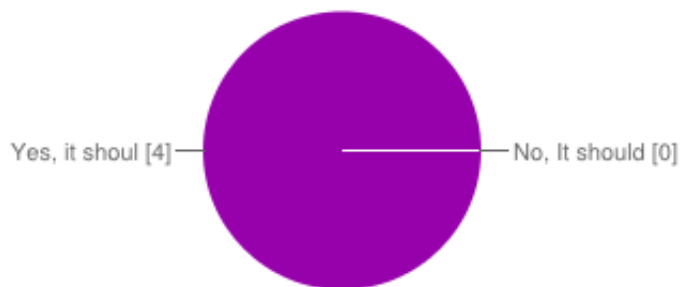
1) Didn't validate user input. User could enter 'fg' and it would process it. However it would mark it wrong and not add it to my guesses list. consider adding this to your .xml file --> `android:maxLength="1"` 2) After a user types a guess they have already guessed before, the edit text field should clear instead it keeps the old character. 3) Once a user has lost, consider locking down the edit text field with this --> `user_guess.setEnabled(false);` where `user_guess` is the view for the edit text field.

i can put any string in the guess bar. i don't know if that is some functionality of the game or not. strings like that do not show up in guesses.

At first I thought allowing more than one character guesses was a bug but I see from the code these kinds of guesses are considered whole word guesses. This is a nice addition but the execution was a little confusing. It allowed guesses for the word that were not the same length as the current word. This made it unclear that when I type more than one character the game thought I was doing a whole word guess. Another suggestion I have is adding an on-click listener to the EditText for guesses. Adding: `android:onClick="Guess"` to the xml file should do the trick.

Nathanael Crabb

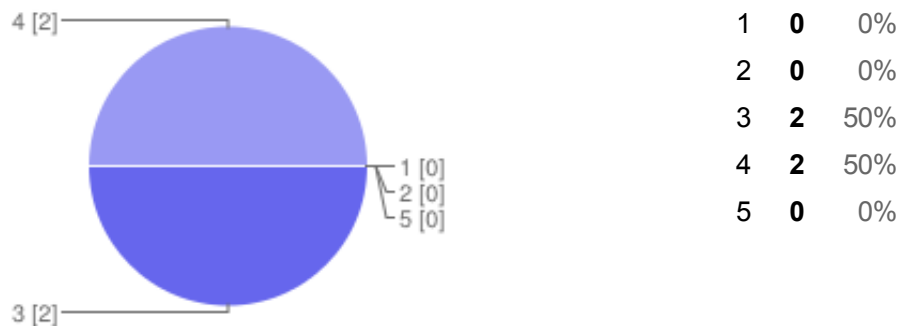
Should the assignment receive credit?



Yes, it should receive credit.	4	100%
No, It should not receive credit	0	0%

Awesomeness

On a scale from 1 to 5 please rate the look and feel of the app



Praise for submitter

Solid design, and fluid gameplay.

I love the fact that you put who wins in two places. One via the message at the top and via a toast. Great work! Very Quick and responsive!

Simple clean layout makes app very user friendly. The addition of rules was a great idea. The app doesn't appear to have incorrect behavior.

Game works well

Suggestions and Criticism

I would suggest taking out the hardcoded strings in TicTacMainActivity.java and adding them into the strings.xml resource. Also a nice touch would be adding a message in the case of a tie. Currently the game just waits for the user to hit new game. Another idea would be to have one ButtonClicked method that grabs the idea of the button that called it. This way you don't have 9 functions that are essentially the same thing. For example you would just change the code from this: `Button button = (Button) findViewById(R.id.SCButton);` To this: `Button button = (Button) view;` To my understanding the view you pass into the function is the view of the button that called it. So all you have to do is cast it to a button and go from there. This way you don't need the findViewById calls because you already have the view. The last thing is completely optional but I think it would be fun to add a score for how many rounds player X and O have won.

There is no message for tie. I would have added a message somewhere (like top right) that indicates who's turn it is...

none

Erica Huang

Should the assignment receive credit?



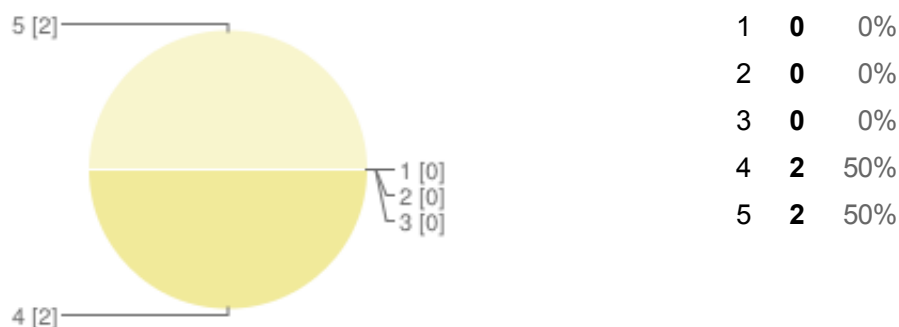
Yes, it should receive credit. **4** 100%

No, It should not receive credit **0** 0%

Awesomeness



On a scale from 1 to 5 please rate the look and feel of the app



Praise for submitter

I really liked how you put a keyboard in front of the user and it showed what they had guessed! That's awesome! I also liked that it changed color based on if it was correct or not.

Awesome layout awesome game.

The layout is nice, the user interface is simple and easy. I really like the keyboard that shows what letters you've already guessed as red and the correct as green. Really cool and great way of handling user input. Nice background image too. As for the code it all looks good. I really like how you put all the hangman images in a global array.

Great color and layout! I like that you made your own keyboard, and that it displays the correct/incorrect guesses.

Suggestions and Criticism

none

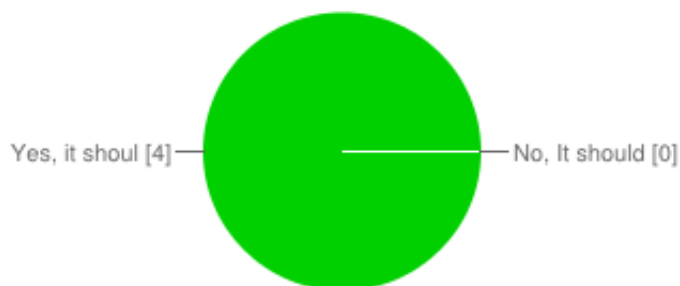
Why not just reveal the word once the user guesses wrong? I liked the background but some of the graphics got in the way of the text...

N/A

The only feature I could see adding would be to display the word after the user has lost so they don't have to hit forfeit to see it after they lost. Not having that doesn't take away from the app very much.

Jaspaul Sohal

Should the assignment receive credit?



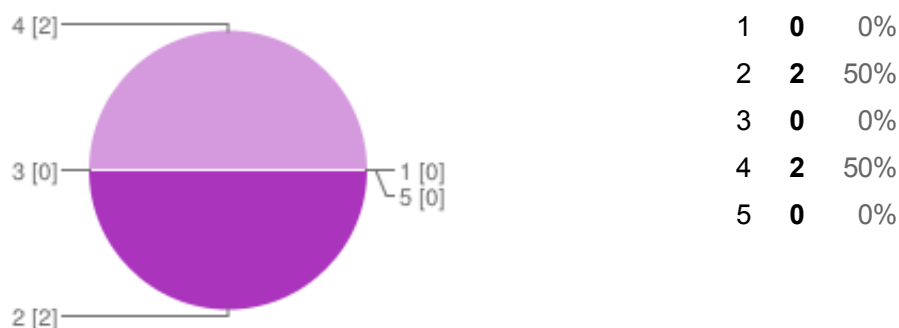
Yes, it should receive credit. **4** 100%

No, It should not receive credit **0** 0%

Awesomeness

No responses yet for this question.

On a scale from 1 to 5 please rate the look and feel of the app



Praise for submitter

Solid design, and fluid gameplay.

functioned well

The app demonstrates good use of Toast messages to inform the user. For example when the game has ended if the player tries to move again it tells them the game has ended. The user interface is clear and there aren't any confusing parts to it.

Good job catching all the game win/lost/tie conditions! Excellent work! Game is very responsive and toasts are very informative. Also great job at locking the board down once a game condition (win/lose/tie) has been met.

Suggestions and Criticism

The layout was poor and did not meet requirements.

In the method checkForWinner I would suggest breaking the if conditions into two lines due to their length. Also instead of saving the button views as b1 through b9 for use in the onClick method it might be better to use the view v and cast it to a button in the call for updateBoard. It would look something like this: `onClick(view v){ validmove = updateBoard((Button)v, p1_turn, board);` If I'm correct that should work.

I would have centered the game since you have the title centered. But that's it!

none