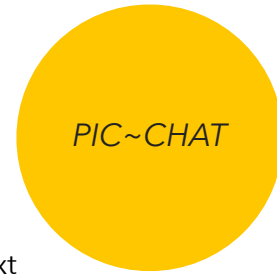




PROJECT SUMMARY



ABOUT: "A picture is worth a thousand words"

This is a messaging application, Instead of conventionally sending text messages it allows you to send images and videos to your friends.

Few Notes before using app:

- ▶ Please use an iPhone 6 device to test, as auto layout hasn't been implemented
- ▶ It would be nice if you had an Gravatar account(<http://en.gravatar.com>), this would enable your friends to see your "avatar".
- ▶ In the case the logout button doesn't seem to load/appear please restart the app and it will work.

PROJECT FILES:

There are 2 project files, they are small variations of each other:

- ▶ The original pic-chat: make-friends, send and receive images and videos.
- ▶ This pic-chat app deletes the message automatically once viewed

THINGS I WOULD ADD IF I HAD MORE TIME:

- ▶ Everything from the examples list except embedded media(no utility for youtube media in this app).
- ▶ Would add text message support.
- ▶ More interactive UI (essence of social apps!)
- ▶ Auto Layout
- ▶ Managing the keyboard (in smaller phones the keyboard covers the text boxes) .
- ▶ Somehow refactor the UI to look slightly different everyday(ex: different background of login screen everyday)

THINGS I WOULD CLEAN UP/REFACTOR IF I HAD MORE TIME:

- ▶ I would fix the checkmarks bug, note that we are not able to deselect/unfriend the user (biggest loophole facepalm!). I have the code but the sequence but there is some issue with the sequence of execution due to which some statements are not being executed as desired.
- ▶ Check if I am making any asynchronous calls, and make them synchronous. For example: I am obtaining the Gravatar images of users and having it as the person's image, initially I had made asynchronous calls, but later had changed it. This decreased the time to populate the friends list from 15s to less than 2s.
- ▶ Organize my code better so that when I increase the features (MVC model), I don't have issues debugging.
- ▶ In general I would try to work on the app until I feel that I would use this app over the others in the store:)

SOLUTIONS TO ABOVE STATED BUGS:

Although I ran out of time to fix the major bugs in the app, I have learnt the ways to fix most of them.

- ▶ The Keyboard issue is very common, The best way to fix it is to use the `TPKeyboardAvoiding` API and integrating it with a scroll view.
- ▶ Unable to delete friends: Implementing the checkmarks was just a simple sequence of for loops and if else statements, the order in which the execution is taking place needs to be looked at keenly.
- ▶ The current user also appears in the friends list: the idea here was that I might want to send a picture to myself and that's why I had done this. But I have realized that this is useless, and can be undone by having an if block to filter the users.

THINGS I LEARNED/USED:

- ▶ Adding progress indicator-pull down the inbox screen, refreshed the list (like Mail)
- ▶ Using `TPKeyboardAvoiding` API, far more efficient than the conventional way listed in iOS docs.(managing the soft keyboard-keyboard covers the login screen textboxes)
- ▶ Grand central dispatch - Gravatar images added synchronously.
- ▶ Processing videos and images (exposure to `UINavigationController`, `MediaPlayer` framework, etc)
- ▶ Storing and retrieving data from backend system
- ▶ Blocks and `TabBarController`s

CONCLUSION:

Goal: My intention was to make the app as complete as possible with minimal bugs and to have a clear and concise code in how much ever little progress I might make.

- ▶ Thanks for such an interesting app, I will work harder on it and try to make it as useable as possible.
- ▶ If possible please let me know in case my code was unclear/not structured well or I took too much time and achieved less. I will try my level best to improve because "products are my passion."

"Why text when you can click!" ~ PicChat

Hope you liked it. Thank you:)

~ Lohit

