Last Update :- 20-Apr-2022

iOS

Created by - Mohammad Mugish

GOALS

1. Branching Convention
2. Tickets
3. Merge Request and Code Review
4. Code Quality
5. Learning new skills

1. Branching Convention

From the master branch we will create a Feature Branch with this naming convention -

- “Feature/FeatureName”

For example:-

We picked one new Feature called Profile, so first we will create one branch

- “Feature/Profile”

2. Tickets

The Product team will create Feature feature-based ticket, once the feature is assigned to the respective developer. The developer will create the Subtask for that. Each Subtask will be followed by one branch while integrating. Once the Subtask is finished we have to create PR pointing to the particular Feature branch and that particular MR should be linked with the Ticket.

State of Tickets:

1. To Do
2. Under Development
3. Ready for Review
4. Merged
5. Ready for QA
6. QA Failed
7. QA Done
8. Released
9. Backlog

3. Merge Request and Code Review

Acceptance criteria for a merge request

1. The merge requests should be attached with Jira tickets for a particular feature or task
2. The merge request should be as small as possible but shouldn’t be more than ~300 lines of code **Hint:** Break your Jira tasks into very small tasks that can be done and the code can be reviewed
3. The code should adhere to the coding principles

4. Code Quality and Naming Conventions

Code quality will determine how fast the team will be able to ship the new feature

Following are the questions you can ask to pass the code quality measurement before creating a merge request

1. Can your code be easily read by even beginner developers?
2. Is your code extensible, can it be edited by developers who aren't the original author? Is your code easy to maintain?
3. Is your code well-tested for quality and bugs?
4. Does regularly updated documentation accompany your code?
5. Is your code refactored regularly to reduce the problem of technical debt?

6. It’s the responsibility of the development team to test and produce well-tested and good quality software and in order to achieve it, we should keep changing our hats from developer to tester, customer, etc.

5. Learning new skills

Acquiring skills is one of the crucial habits that any individual can form

1. Skills that is required to improve the efficiency and performance of the Application

- SwiftUI for UI Designing
- Combine for Reactive support

2. Skills to write quality code

- Clean code by Bob Martin - book
- Refactoring by Martin Fowler - book
- Clean code fundamentals by Bob Martin - A free video course https://www.oreilly.com/videos/clean-code-fundamentals/9780134661742/

3. Skills to apply agility to software development

- Scrum: The Art of Doing Twice the Work in Half the Time - Book