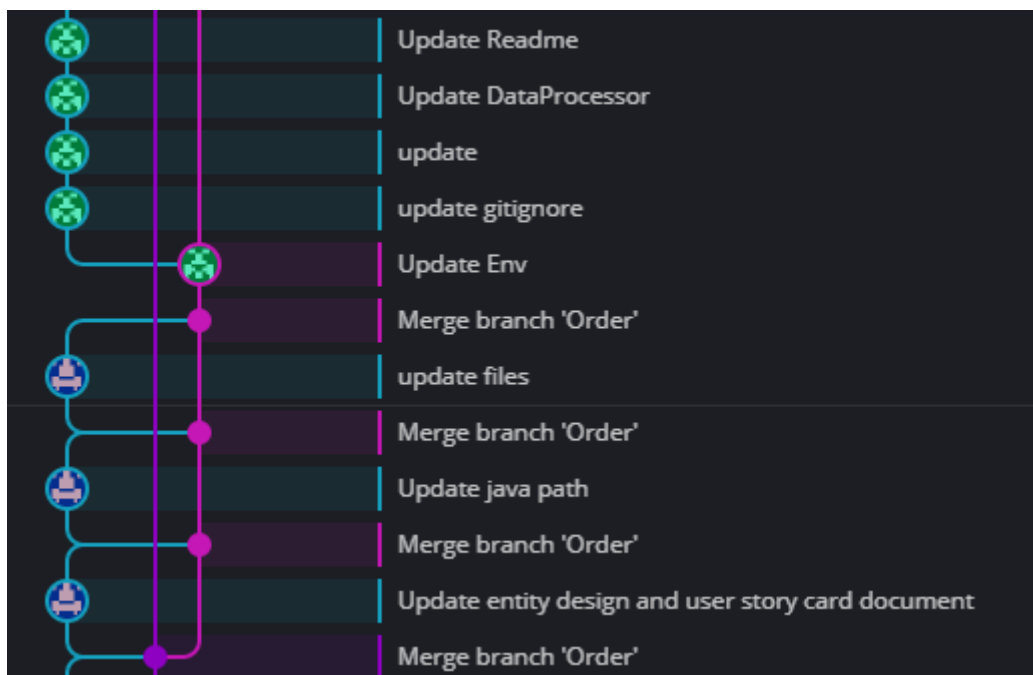


## Choice of tool

The current special situation forces our group to collaborate online, we have thought about using IM applications to share files, but end up with using professional tools. SVN and git is our main option, considering the different learning curve and fault tolerance of these two tools, we decide to use git for collaborative development and version management. Building a git server for our own is never a option, so we host our project to github remote collaboration repository, and in order to simplify the operation we choose gitkraken as a graphical tool for git.

## Project collaboration

One of our group member is hosting the github repository, and only maintain a master branch in remote. We require every group member to create a new branch in local when they start to develop their own part of function, who also needs to merge their work to local master branch then push them into remote master branch when they finished. In this way, we can iterate our projects step by step and generate development logs automatically with the help of gitkraken.



## Decision making

We host a online meeting on Friday or Sunday (depend on the schedule of team member) every week to discuss the exsiting problems and specify requirements. Our group member have multiple ways to feedback the their problem, we can discuss technical issues in Wechat or design problem through github message board. Our group leader will collect those problems and share with other member at weekly meeting, then decide the next iteration jobs together.

## Planning, estimating and adapting

For the aspect of planning, we made plans based on the principle of *Arrangement for next iteration must be clear, for further iteration can be concise*. We have made a story list in the beginning that contains most arrangements for the project, which is finishing the noddle select and order functions. The GUI part of work was in developing at the same time since we need a interface to do human testing. The member function is the last because it is a kind of independent system.

For the aspect of estimating and adapting, we firstly estimate whether the function was implemented, if so then we test it with some boundary cases to check its fault tolerance and stability. In the end we consider the coupling and cohesion of our project, which is means like, If a part of code is over coupled, we will decouple it before the next iteration. The main method we use to adapting the change is to divide every iteration into three step: fixing previous problems, finishing temporary job, creating extension interface. Step one is the way we adapting the changes, step three is the way we leave space for changes. By using this method, we have guarantee that our project can run correctly in each iteration.