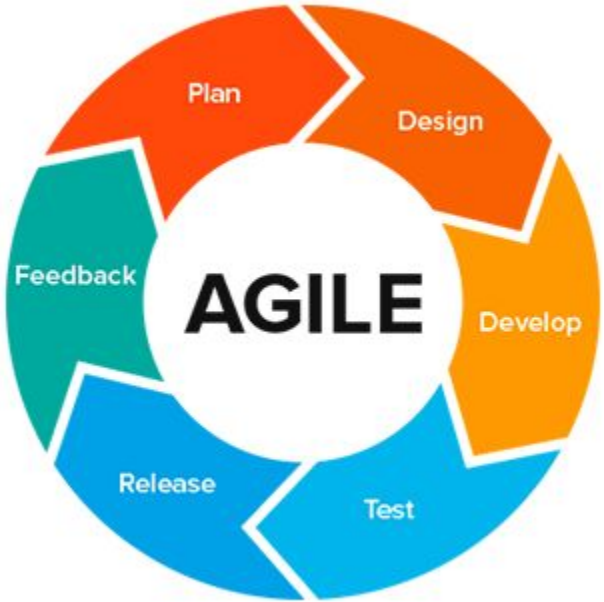
Development Methodologies



What is it?

Almost sure my definition will be incomplete, but nevertheless - it is just an approach with set of techniques to organize work. They can be applied not only to program development

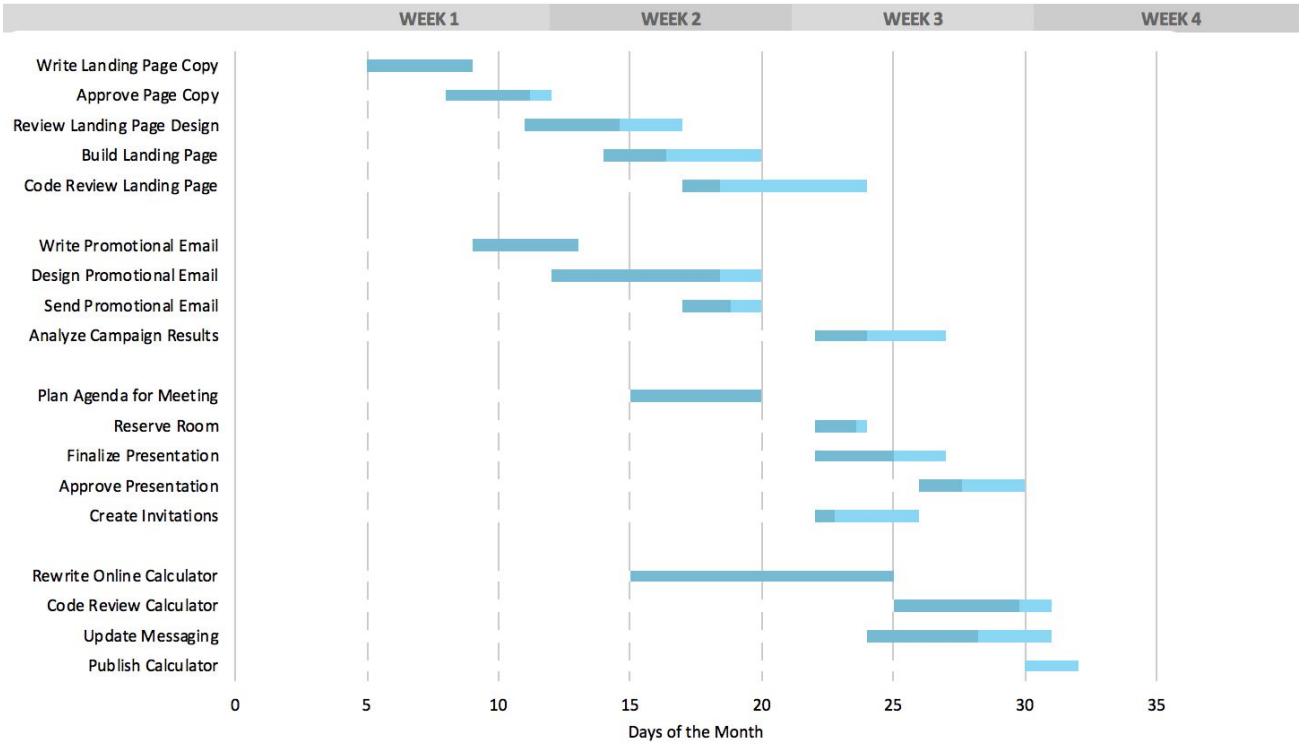
Next you will see a mix of development models and methodologies to illustrate several organization approaches

Organization approaches

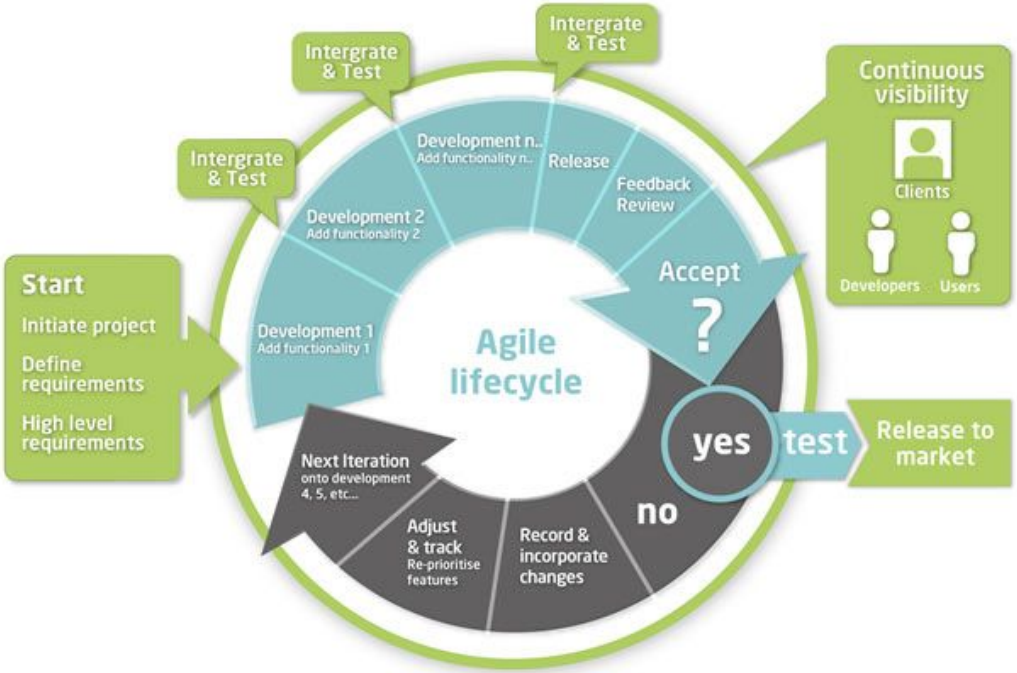
Basically could be divided into 2 groups

- everything (at least main things) is hard planned - usually we know what functionality we desire and when we need work to be done
- we have not so much plans - aim is somehow defined, but not very clear

Constant plan



Changing requirements



Usually traditional planning where everything should be included is opposed to agile approach, where you adapt your program to changing environment

The latter approach looks more viable in current situation. Or maybe it is always better

Traditional planning disadvantages

- It is hard to plan everything, usually something somehow change it
- Aim can change in the time of the development
- It is linked with blurred vision of your customer who don't know what is he want

Agile disadvantages

- Some people think that it is a silver bullet
- Some people are obsessed with rituals (ye, organization techniques can be treated so)
- Development can go ad infinitum (well, not a disadvantage from some point of view)

Agile methodology

Thought to be created in Toyota production where instead of regular conveyer workers can stop it to check correctness or fix errors

There is a lot of branches like scrum, kanban, extreme programming and so on. Some of them are more similar, some less, but they are trying to make development more efficient

Scrum

One of the most popular methodology. Many of its items looks pretty sensible!

- Select Product Owner - person with the clearest vision of the aim
- Form a team (3-9 members) with all necessary skills to reach the aim
- Select Scrum Master - organizer, who destroy obstacles
- Create a Backlog - list of features which need to be brought to existence in order to reach the aim; Product Owner decide what should be here and priority of the items
- Backlog estimation - team estimates points for every item in terms of completability, required effort and gain from it complete, after that items are sorted with the most important on the top

Scrum principles continuation

- Plan sprint - sprint is 1 work iteration - all together decides how much can they do in the iteration and fix it
  - sprint length is 1-4 weeks
  - items for sprint are taken from the top of backlog
  - how much is defined in terms of points (this sum is deduced from the previous sprints) - it is also called team velocity
- Work progress visualization - usually done via table with To Do, Doing and Done columns, containing items in progress (and they move from to do to done); also visualization can be achieved via burndown plot - left points vs time

Last principles

- Daily meetings - less than 15 minutes for the following questions
  - what was done on the previous day
  - what will be done today
  - what hinder work
- Sprint review - meeting about sprint results, where finished parts are shown
- Retrospective - discussion about improvements of work process

Important conditions

There are several requirements that must be met

- friendly atmosphere
- honesty
- ability to hear feedback

Probably that's all)

Team Task

Split on groups with up to 4 people in group and solve the following problem - create repository with program for music clustering, i.e. it takes folder with songs and produce a scatterplot with songs separated into clusters, where distance represents a similarity. The more you do the better

Intermediate tasks

1. Distribute roles in a group
2. Create backlog
3. Find and learn necessary material
4. Implement backlog tasks and get the result
5. Explain to each other what you have done
6. Create presentations with links to sources
7. Present

Assessment

Each member should know all aspects of the program

Creating dataset

Let's all grab some music for this task - at least 7 tracks from each person. Imho, it will be good to take at least 3 tracks from 1 artist.

If you don't like music you can take other stuff (audiobooks, podcasts, etc)

Create 1 zip archive on the google drive and use it

Hint

Possible subtasks

- music feature extraction
- dimensionality reduction
- clustering
- working together in 1 repository