Software Development Models

Aitor Corchero Rodríguez

Speaker

Agenda

- Predictive vs Adaptive Models
- Software Development

> In previous sessions, we talked about the companies, projects and resulting products. We talked about how the porject is initiated in a company. During this session, we will talk about project planning.

Predictive vs Adaptive Models

> All software development models are categorised in Predictive and Adaptative models. These models represent a way of thinking and solving a problem. As allways, there is not better group of models than others.

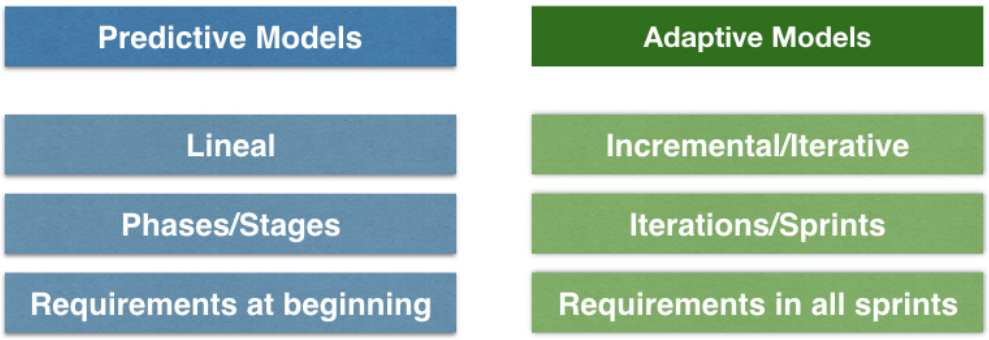
Predictive Models

The first group of models refers to the predictve way of thinking. Predictive models are **Linear** Models in which the project is constantly developed until the end. Predictive models/projects are divided in different **phases or stages** and the project ends once executed all the phases. Commonly, the structure of this kind of projects begins with a **requirements phase** and adter requirements the projects is being developed and tested.

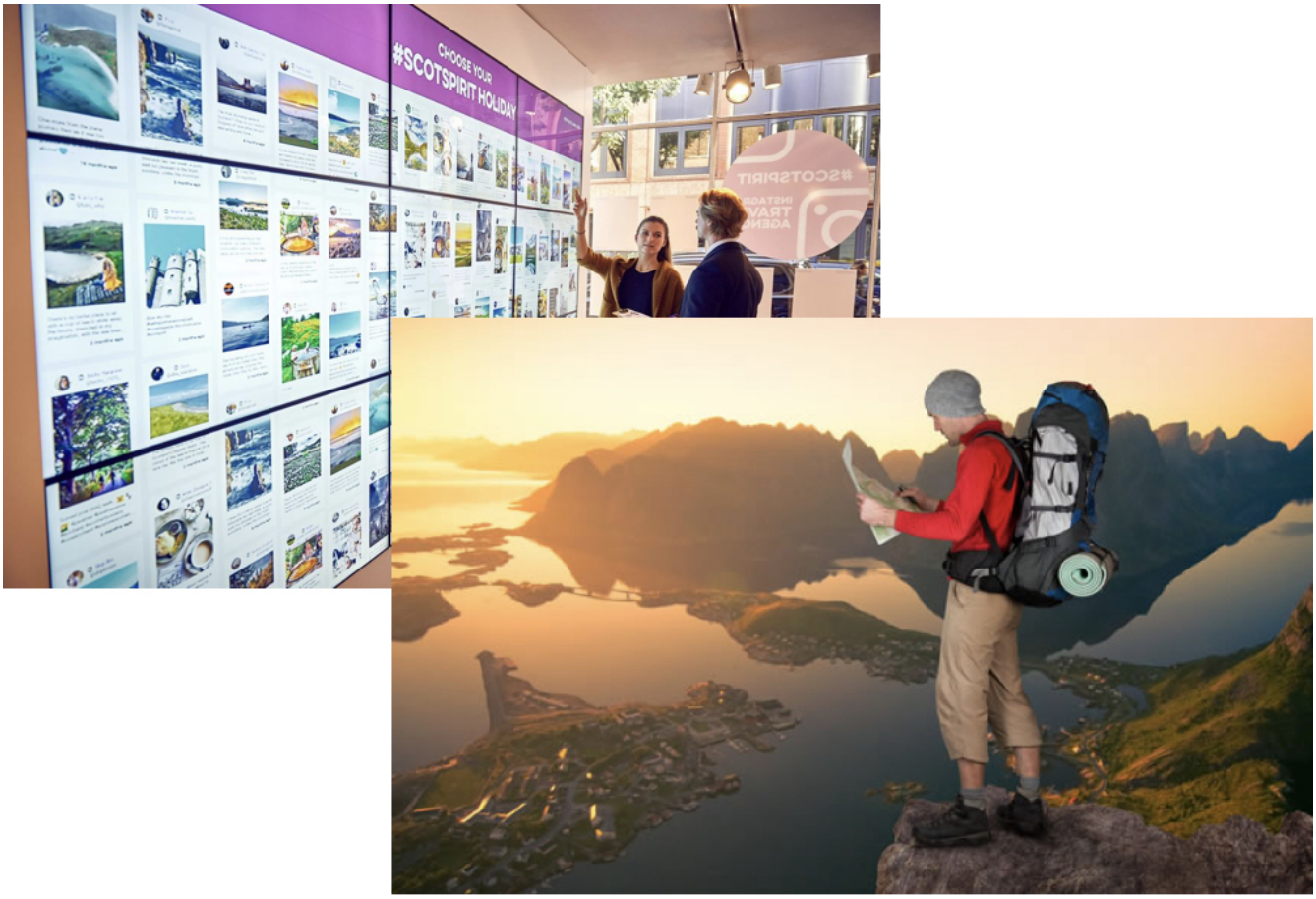
Adaptive Models

As contrary, adaptive models refers to models that iteratively or incrementally improves the product along the lifecycle until achieve the final product. The main feature of this models are the capacity to **adapt** to existing situations. Commonly, this models are divided in iterations and sprints. In each iteration or sprint, a new **functional prototype** is being developed and validated by the clients. Thus, it is commonly that the requirements changes along the project lifecycle (iterations).

Predictive vs Adaptive Models (Summary)

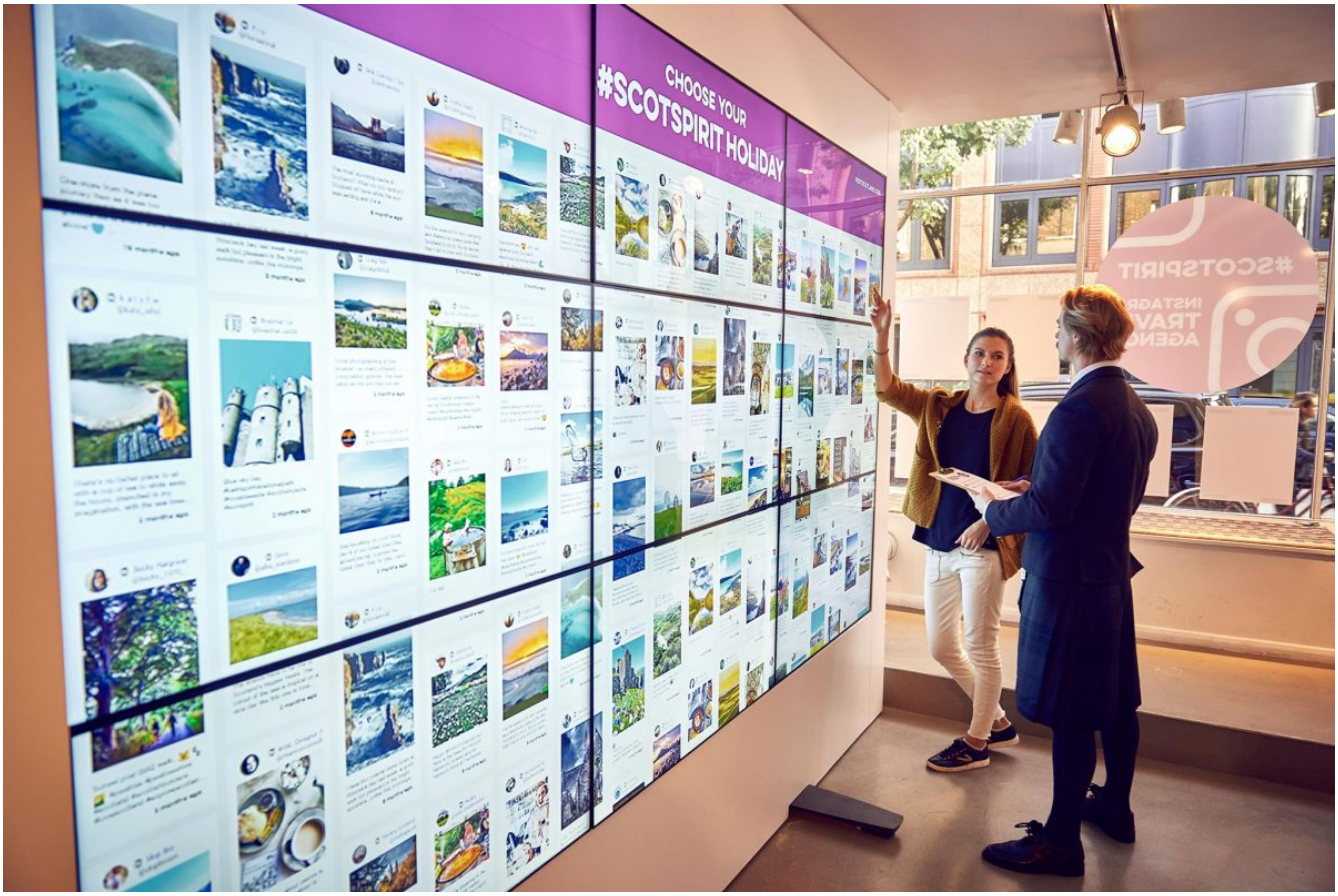


!Predictive vs Adaptive Example



The coming slides will show an example between predictive and adaptive travelers. In this example, we have to plan a travel from Barcelona-London with a budget of 600€ for 3 days. How will proceed each of the travelers?

Predictive Traveller

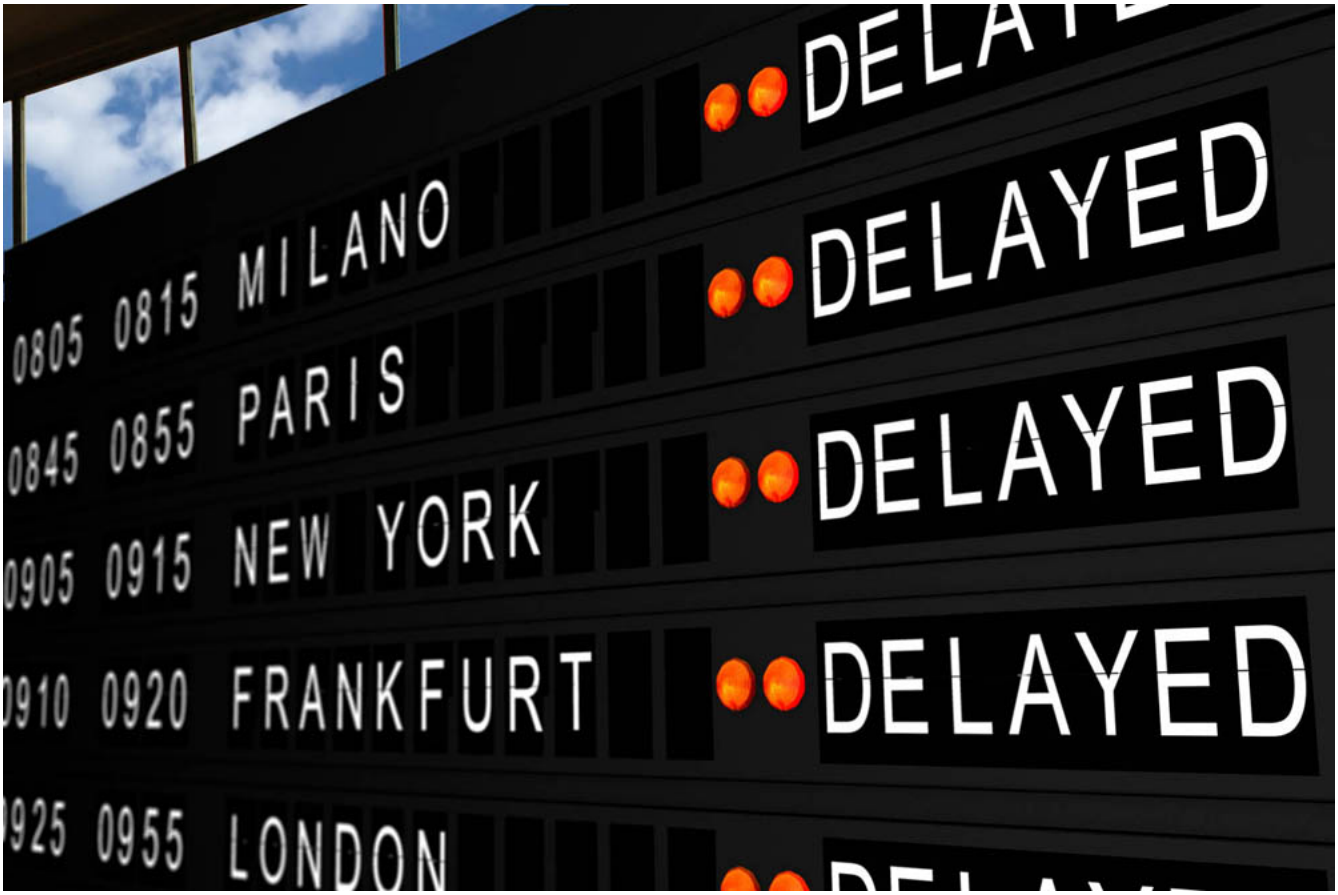


Predictive Traveler Planning

Task	Cost
BCN Flight	€100
London Bus	€60
London Hotel	€300
Monuments	€40
London Flight	€100

The predictive traveler divides the travel in different steps starting from the Barcelona flights, the hotels in London, the monuments to visit and finally, the come back home. Everything is planned and scheduled linearly in time until the end of the travel. What happen if something unexpected occurs?

Unexpected Situation



If some unexpected situation occurs, then, the predictive traveler has to re-schedle the whole plan. It could impact in some unexpected costs and delays at the end of the travel.

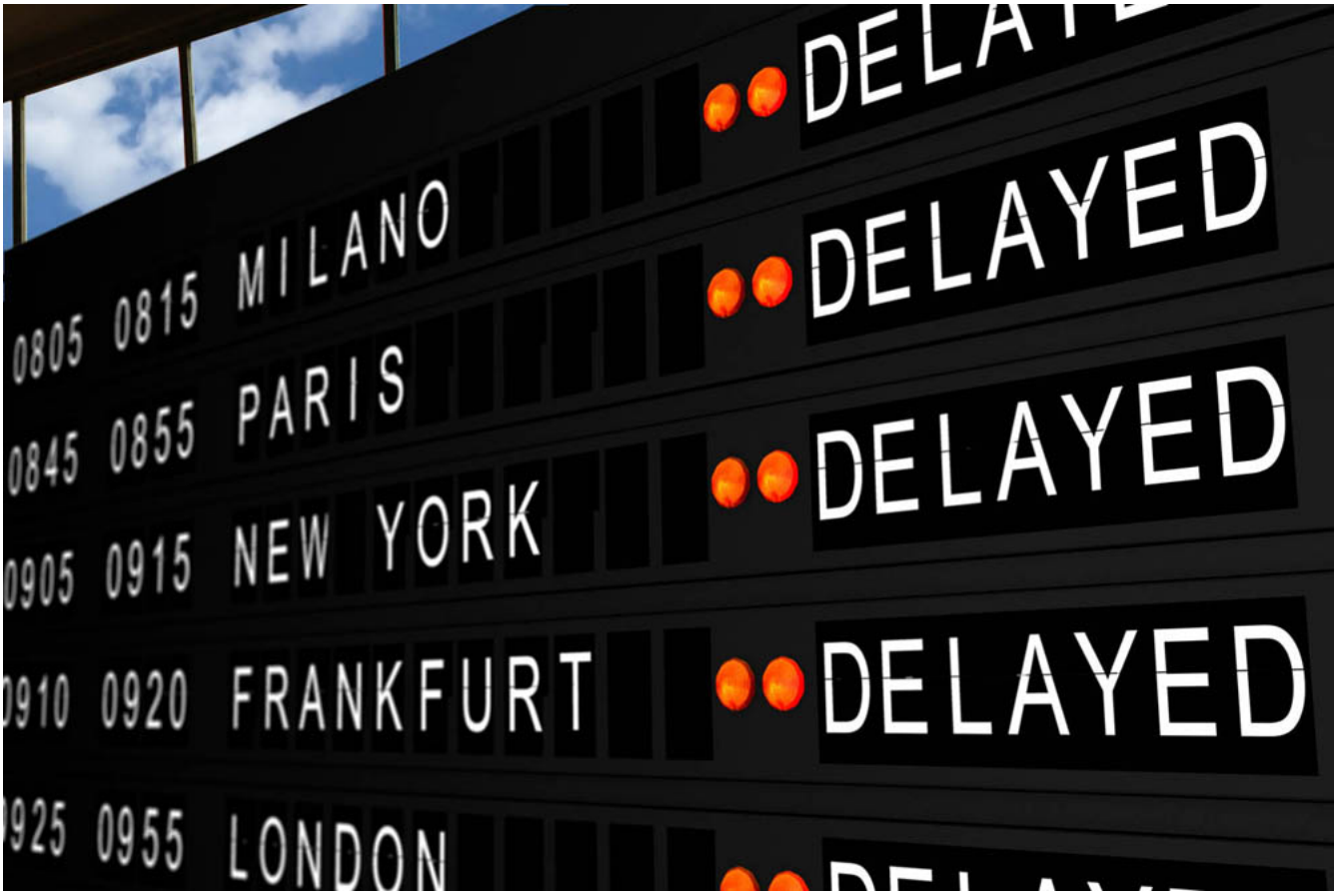
Adaptive Traveller

Adaptive Traveler Planning

Task	Cost
BCN Flight	€100
London Bus	€60

The adaptive traveler only plans the first or two first days of the travel. Then the rest of the days are planned based on the interests of the traveler or the weather. What happen if something unexpected occurs?

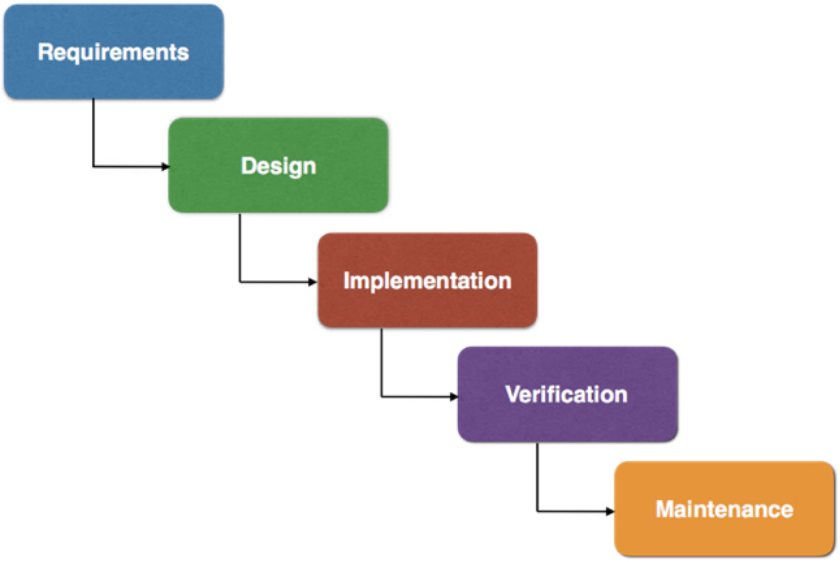
Unexpected Situation



In that case, the asociated costs are minimised to the planned window and makes open the rest of the days to plan anything.

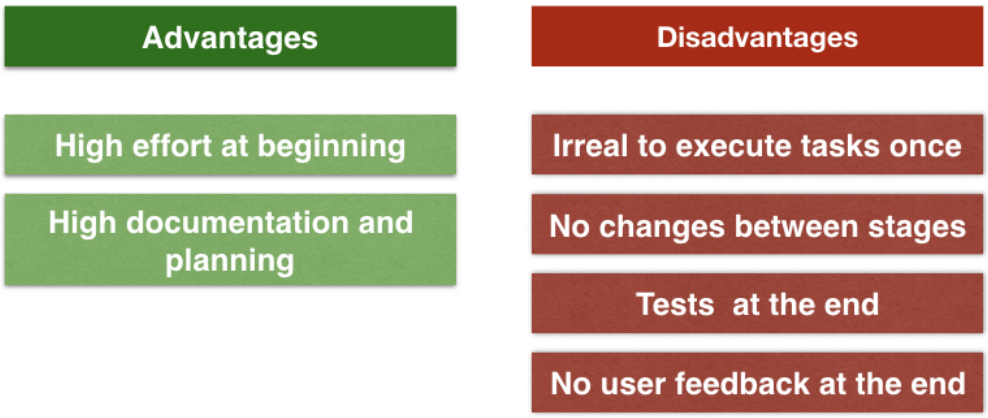
Software Development Models

Waterfall Model



The waterfall model is a sequencial model in which th stages are executed linearly in time. That means, one task do not start until the precedent task ends. Once the task is executed and finished, the task do not execute again.

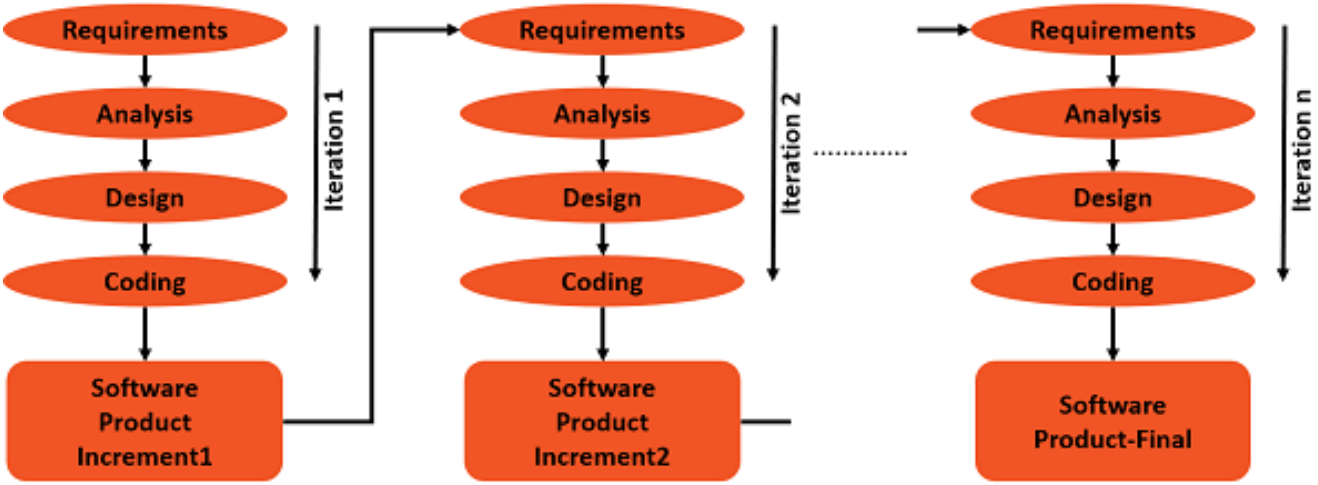
Waterfall Features



Regarding the waterfall model, all the efforts are performed at the begining with exhaustive task in the requirements gathering. Moreover, this model contains high documentation and strict planning. These both aspects are considered as an advantage.

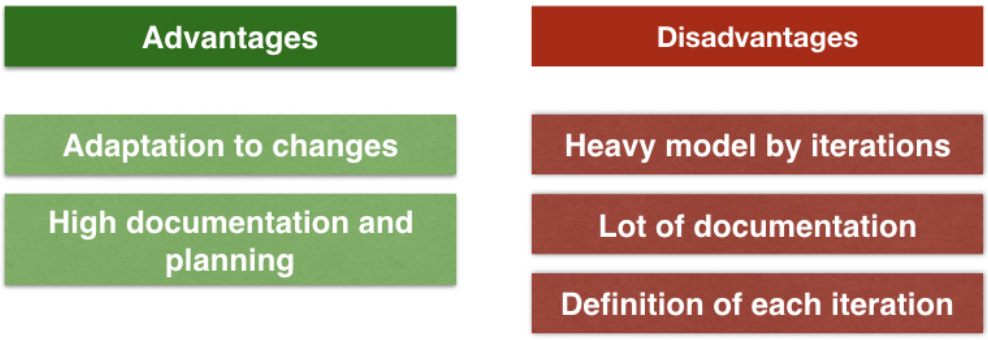
As a disadvantage, this model is quite irreal in terms of strictly execute the tasks. Commonly, it is quite irreal to have all requirements at the beginng. Moreover, it is quite irreal that there is not any change in the project tasks. Finally, this model has the disadvantage of testing the product at the end of the process, putting the response time to error in risks. Moreover, there is not user feedback until the product is delivered (too much risk!!)

Iterative Model



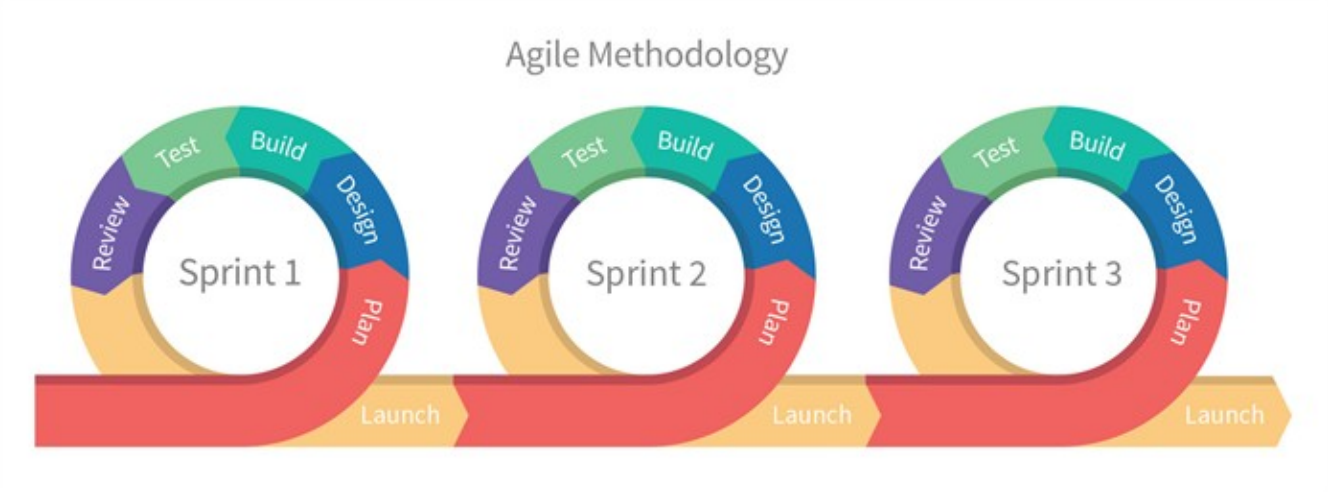
The iterative model is formed by multiple iterations. Each iteration comprise all the development phases (requirements, analysis, design, coding and testing). This method permits to redefine and adjust the work performed along the project lifecycle (at the beginign more efforts in requirements and at the end of the project, more efforts in the testing.)

Iteration Features



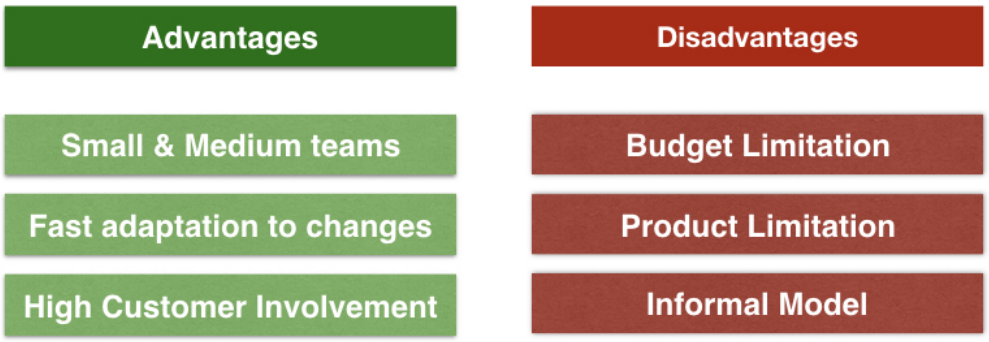
The iterative model is formed by multiple iterations. Each iteration comprise all the development phases (requirements, analysis, design, coding and testing). This method permits to redefine and adjust the work performed along the project lifecycle (at the beginign more efforts in requirements and at the end of the project, more efforts in the testing.)

Agile Model



Agile models starts with the asumption that the changes are unaviodable. Agile methods are adaptable to changes in form of small iterations in which the end-users are continoulsy involved. Thus, the product is evolved considering end-users feedback until the product is finalised and validated by the clients.

Agile Features



In detail, Agile models are ideal for small work teams due to their informal management. Despite this, agile has been implemented in enterprises with high sucessful ratios. Moreover, agile is totally adaptable to changes and customers feedback due to the division of the product development in different and short-time sprints. In each sprint, we create a **functional prototypes** that passess the corresponding test. This **functional prototypes** accomplish several products functionalities that at the end of the sprint are validated by the client. So, the customer involvement is a huge benefit for creating a product.

The constant implication of the clients could be a constraint in terms of defining the final scope of the product (the clients always wants more). This limitation in the scope also is aligned with a limitation in the budget (the clients wannt lot of functionalities at lower prices). Finally, when implementing agile models, we need to take care of teh documentation. In agile there is not so much documentation but we need to elaborate strictly needed documentation for product understanding (specifications).

How to select a software development model?

Feature	Waterfall	Iterative	Agile
Type of Project	Small/Medium	Medium/Large	Small/Medium
User Interaction	Low	Medium	High
Project Adaptations	Very Low	Low	High
Requirements Adaptation	Very Low	Low	Very High

Be ready for plan your project.