‹　**Hiring**

Feature Driven Development (FDD) Processes and Comparison To Other Agile Methodologies

Lea Karam
July 19, 2017 • 15 Min Read

This article originally appeared on Apiumhub and has been republished with permission. Find out how to publish your content with Upwork.

Not many talk about FDD, and one may say that Extreme Programming, Scrum and Test Driven Development are definitely the most popular Agile methods at the moment, but here at Apiumhub, we also value FDD quite a lot.

STARTING OFF WITH THE OBVIOUS, WHAT IS FEATURE DRIVEN DEVELOPMENT?

First, I would like to mention that FDD was created by Jeff Luca in the late 90's. He was actually trying to provide a software development solution to a bank. FDD is a development process that, as all agile methodologies, is iterative & incremental with the objective of delivering working software. FDD mixes best practices that are all driven by what is important to the client. This means that the developers focus on the features that the client values, the functions they expect. Let's say that with FDD, features are as important as user stories are for scrum. It's what will help developers when it comes to planning their work. As I mentioned earlier, Jeff Luca was the creator of FDD. He proposed a solution which is

A MORE DETAILED OVERVIEW OF THE 5 PROCESSES OF FDD

Once we will go over the whole processes, you will quite quickly realize that the Chief Programmer had a very important role in Feature Driven Development. Almost comparable to a lead developer, the Chief Programmer needs to have technical skills as well as leadership skills to be able to lead a cross-functional development team. Another very important role is the Domain Expert as he has very similar responsibilities as the Product Owner in Scrum, although not totally the same.

1st process: Developing an overall model

In this first process, FDD pushes teams to build an object model of the domain problem. Differing from others, FDD modelling is a cross-functional, iterative & collaborative activity. The team members (development, domain experts & chief programmers) work together to compose a model for the domain area and are guided by a Chief Architect. The idea is to have different teams proposing different models and later on, after getting reviewed, choose an option, or mix them up. Finally, the domain area model will be merged into the overall model. It's actually a great way to start the project as it enables the team to get a strong understanding of the project as well as a solid communication.

2nd process: Building the feature list

Here, you could compare the features list to the product backlog in scrum, and the feature would be some sort of user story. After having the overall model ready, based on the knowledge got during that phase, we will have to identify the features which are valuable to the client and which will basically guide the project. Features shouldn't take longer than two weeks to be completed, and if they do, then it should be put into more than one feature. They are usually expressed as an action, result & object.

3rd process: Planning by feature

In the third phase, as its name says it, its more or less about planning in which order the features will be implemented, it's about organizing. Feature sets are then assigned to programmers. Obviously while planning we take into

As during all the processes, we use the knowledge we got from the first modelling process. The chief programmer takes responsibility to select a group of features that should be developed next. He will also have to determine the domain classes that will be involved. After the feature team is formed, they all start working together in order to get the job done, where the domain expert will be in charge of analyzing & designing a solution to each feature.

5th process: Building by feature

Once the domain expert is done and based on the work done in the design by feature process, the class owners will have to implement all the items that are necessary to be able to support the design. Therefore, we work on the code that has been developed and with unit test it and inspect it to ensure that it is all correct and approved by the chief programmer that will then give the ok to start building.

What about comparing FDD, XP programming & Scrum?

So we use Scrum, we use XP proramming, FDD and more, so I think it can be interesting to make a brief comparison of those 3. Feature Driven Design has a bit of eXtreme Programming as well as a bit of Scrum but adding to them Domain Driven Design techniques. What is great is that it is very easy to work in large teams using FDD. It's actually extremely scalable.

In case you're interested, here's a comparison of Scrum, Kanban & Scrumban.

Communication; verbal & written

We all know that Agile methodologies have a strong focus on communication between the team and the rest of involved individuals. In XP programming & Scrum, documentation is important but it doesn't push the team to put a strong effort on it and pushes them more towards having verbal communication with the rest of the people implied in the project. With FDD it's a bit different because they actually believe that documentation should be quite worked on.

User centric approaches?

working software is always tested by the user. In Feature Driven Development, the end user is also involved in the process but in a different way, it's actually while reporting. And in Scrum, the end user is not really involved, it's the product owner that is seen as the end user. The sprint's length. This is not a general rule of course but in general as we mentioned for FDD, the shorted the better. Let's say that a sprint would be between 2 & 10 days. Which differs to Scrum that is between 2 & 4 weeks and XP programming that can last up to 6 weeks!

What about the meetings?

As communication is important, obviously, meetings are important with Agile methodologies. With Scrum & XP programming, there are the daily meetings where all the team members are involved and where they talk about the project and decide together how the project should go on. Those meetings are in general quite informal and quick. With FDD its quite different because in general the information will be communicated via the documentation.

How big can the project get?

When choosing Agile methodologies, it really all depends of the project requirements. For example, for small projects that are not complex, you could easily go with XP programming. While Scum & FDD would be recommended when it comes to software projects that are more complex and that are bigger.

SO WHAT IS SO GREAT ABOUT FDD?

- FDD is amazing for big projects and is actually quite scalable and prone to get achieve success.

- FDD is very effective in helping with complex projects that are in a critical situation.

- The 5 processes mentioned earlier help when it comes to getting new members to join the team, specially in short periods of time.

client. You can easily identify errors in the source code of the features.

- All along the processes you have a high visibility of progress and results due to the fact that there are frequent progress reporting that are made at all the levels of the project

- The first process, developing the overall model makes us have a deep understanding of the scope and the context of the project.

- The fact that you have a deeper understanding of the requirements and the expectations, that we do small iterations and build small parts, one by one, implies that the risk is really reduced. Less unwanted surprises!

This story was submitted by Lea Karam and does not constitute the views or opinions of Upwork.

Upwork is a freelancing marketplace where businesses of all sizes can find talented professionals across multiple disciplines and categories. If you are a business and are looking to get projects done, consider signing up!

Join Upwork!

Project Management   Agile

Share this article   f   in   🐦

Related Articles

11 Tips to Optimize JavaScript And ...

Igor Chishkala
May 26, 2019 • 32 Min Read

C# vs. C++: Which Language is Right...

Jennifer Marsh
May 25, 2019 • 10 Min Read

Writing a Job Descri Find a...

Carey Wodehouse
April 12, 2019 • 12 Min Re

Subscribe for Weekly Updates

Upwork is the leading online talent solution. **Learn More**

Or follow our **RSS Feed.**

COMPANY ⌄

RESOURCES ⌄

BROWSE ⌄

FOLLOW US

f   in   y   ▶   ⊙

MOBILE APP

© 2015 - 2019 Upwork® Global Inc.