Jennifer Cafiero

SSW 555 Homework 9

The seven roles in FDD are:

* Domain Experts – People in this role act as the voice of the customer. The domain expert(DE) knows what the customer wants from the new horn honking feature and the DE will represent the customer in any meetings with the team members. This person will be able to provide the necessary car or horn information to the Chief Programmers and Class Owners. The corresponding role in Scrum would be the Product Owner. Both the domain expert and the product owner act in the customer’s best interest and either can be approached for questions regarding the customer’s needs.
* Project Manager – The project manager is the administrative lead for the project. The project manager is responsible for eliminating administrative roadblocks and ensuring there is appropriate documentation. The project manager would communicate with executives of Driverless Cars and provide them with updates on the team’s progress. The responsibilities of a project manager coincide with those of a Scrum Master, where both roles’ main goals are to eradicate impediments to the team’s progress.
* Chief Architect – The chief architect is responsible for the overall design of the project. The chief architect (CA) is responsible for making the higher level decisions for the scope and direction of the project. The CA is responsible for deciding what the program for the horn feature would look like from a UI perspective, and would determine how to integrate it into the car. This role corresponds to Scrum’s Product Owner who is also responsible for making many of the important product decisions,
* Development Manager – The development manager is responsible for team management. The development managers decide which developers will be on what feature team and what feature each team will specifically work on. In the Driverless Cars scenario, the Development Manager will assign different features to each developer corresponding to their skillset and background. This role performs similarly to how the development team in Scrum selects specific items from a sprint backlog for the next sprint.
* Feature Teams – The feature teams are temporary groups of developers that are implementing the features. Each feature team is consisting of 2-5 class owners lead by a chief programmer. These teams correspond to scrum teams consisting of programmers being led by more experienced programmers. A feature team for Driverless Cars would be to program a sensor to automatically honk if a neighboring car comes too close to the autonomous vehicle. The chief programmer would be responsible for leading the team and the class owners would perform most of the development for each feature.
* Chief Programmers – The chief programmers lead teams in the design of features. The chief programmers hold most of the responsibility for an individual feature. The chief programmer in Driverless Cars would be primarily responsible for the aforementioned feature of the horn honking when a car gets too close. This programmer is the one that the lesser skilled developers should go to if they have any questions about the feature being developed. The more experienced developers on a Scrum team would take on a role like this.
* Class Owners – The class owners are the developers who own individual classes in the project. The class owners take full responsibility for each class of code that they write, through the build and execution steps. The class owners are related to the Scrum team, in the fact that both parties perform most of the programming.

The advantages of distinguishing each role in FDD is that it affords the development team more accountability and organization. Each of the seven roles in FDD has clear responsibilities and everybody involved is aware of the expectations of each other. The structured roles of FDD make it easier for teams to assign and delegate new responsibilities. A disadvantage of FDD is that it requires more “heavy lifting” than Scrum’s flat structure. FDD involves more structure and planning so there is less flexibility in the development cycle and the project progresses slower. Still, I would recommend FDD to the Driverless Software. I think the accountability that FDD can provide to this project would be beneficial because this project is so important. Using FDD would make the team members’ roles more clearly defined and allow for the project to be completed smoothly.