Chapter-6

CONCLUSIONS AND FUTURE ASPECTS

After completing this research, it is concluded that there are many traditional existing models for developing stems for different sizes of projects and requirements. These models were established between 1970 and 1999. Waterfall model and Spiral model are used commonly in developing systems. Each model has advantages and disadvantages for the development of systems.

In the traditional software development methodologies, the lack of user input, incomplete requirements, and changing requirements are some of the major reasons why software systems do not deliver all their planned functionality on schedule and within budget. In the field of software engineering, several problems became apparent in the traditional software development methodologies; and this is due to the inflexible division of a project into separate stages [2]. Because software developers made commitments early in the development process, it is difficult to react to changes in requirements. Due to all these problems agile is prominent nowadays.

Agile projects have no fixed price or fixed schedule and projects are open-ended and evolve as requirements changes. Therefore it becomes harder for the manager and customer to accept this technique as customers would rather know the total cost of the project and overall project schedule beforehand. On the other hand it can be pointed out that agile and fixed price are not mutually exclusive.
With regard to agile it is found that the migration from plan-driven to agile development is beneficial. The agile and lean practices introduced in this thesis are perceived useful by the practitioners. The practitioners identified concrete decisions in which the lean solutions could support them. In addition, the Lean solutions allowed identifying concrete improvement proposals to achieve Lean software process.

So, keep the client's needs at the top of mind. All teams should be hearing "the voice of the customer" as often as possible. Involve clients in the development life cycle as much as they're willing to be involved. At all levels of the organization and external client base, constant communication and feedback are truly valuable.

According to finding, most view ASD favorably due to improved communication between team members, quick releases and the flexibility of designs in the agile process. On the other hand, developers worry about scaling agile to larger projects, attending too many meetings which contribute to excessive overhead, and experiencing difficulty in getting management to buy into ASD methods. The results of this thesis contribute to understanding of how ASD methodologies are being implemented in the workplace.

An interesting fact that is noticed is that a company mostly developing small projects is enjoying a good amount success in using agile methods. It is also observed that the companies are not really being encouraged in terms of using a variety of agile methods. This is because of lack of sound proficiency in this field. It
is therefore important that empirical research and investigation should be carried in this field. It can be understood that agile methods are highly practical oriented. Application of different practices of agile methodologies differs from company to company. The developers involved mainly stressed following reasons for adopting agile methods: adaptability to change, short time frames of releases, continuous feedback from customers, high-quality and bug free software.

The future work form this analysis is outlined below:

- **Scaling**
  
The team size is typically between 2-15 members for small scale projects and for medium scale 15-50 and large scale projects 50-200. Teams for large scale projects at Microsoft and other companies can be much larger than 200-500. It can be investigated how agile can be adapted to work for these large teams.

- **Coordination**
  
  In large companies agile is not adopted simultaneously by all teams. It can be planed to study how agile teams coordinate dependencies and deliverable with non-agile teams.

- **Empirical body of knowledge**
  
  Collaborate with people in the empirical community to replicate these studies in industry and academia to build an empirical body of knowledge about the various facets of ASD.
• **Product and process measurement**

  Measuring product measures (LOC, complexity, failures etc.) and process measures (productivity, requirement volatility etc.) for ASD projects to compare against non-ASD projects. This would enable us to identify the proper contexts in which ASD should be used.

• **Tools and resources**

  It can be identified that at which areas tools for ASD can be used to improve communication, quality, scheduling and estimation.