**2.4 Software Process Model**

Now time it is difficult or impossible to predict how computer based system (e.g. web based app) will evolve as time passes because market condition change rapidly, end users need evolve, new competition threat emerge without warning, a usage scenarios might have to be modified, a list of functions may be extended which means new requirement should be added as needed any time.

In our software process we will use Agile model with different methods that involved in this model. Among those methods we will use combination of extreme programming, Dynamic systems development method together with other methods because of the above and following reason.

Agile software development model encourage our project to:

* Add any required functionality and modify scenarios any time as needed
* Work in Small, highly motivated project team
* Use informal methods (less formal and reduced scope)
* Customer satisfaction and early incremental delivery of the software
* Edit written specifications
* Make software engineering work products minimal
* Overall development simplicity
* to deliver working software quickly
* Speed up or bypass one or more life cycle phase (based on iterative approach to software development)
* Active and continuous communication between developers and customers

Agile software development model uses different methods. Among those methods XP and DSDM suits for our project.

**Extreme programming method**

We use XP in our project because this method suits for the projects those have the following behaviors

* For small to medium sized teams developing software with vague or rapidly changing requirements
* Coding is the key activity throughout a software project
* Communication among teammates is done with code and necessary documentation is done by decision as its importance and our project have these behaviors.

Among practice of XP that we prefer using more in our project are:

* Small releases: put a simple system in to production, then release new versions in very short cycle
* Metaphor: all development is guided by simple shared story of how the whole system works
* Simple design: system is designed as simply as possible (extra complexity removed as soon as found)
* Refactoring: programmers continuously restructure the system without changing its behavior to remove duplication and simplicity
* Pair programming: all production code is written with two programmers at one time
* Collectively ownership: any one can change any code anywhere in the system at any time (off the developers)
* Continuous integration: integrate and build the system many times a day, any time a task is completed
* On-site customer: a user is on the team and available full time to answer questions
* Coding standards programmers write all code in accordance with rule emphasis communication through the code

**Dynamic system development method**

The Dynamic Systems Development Method is an agile software development approach that provides a framework for building and maintaining systems which meet tight time constraints through the use of incremental prototyping in a controlled project environment.

* Applies a frame work for RAD and short time frames
* Majority of requirement can be delivered in a relatively short amount of time

DSDM principle helps us in the following way

* Active user involvement imperative
* Teams empowered to make decisions
* Focus on frequent product delivery
* Iterative and incremental development- to converge on a solution
* Requirements initially agreed at high level
* Collaboration and co-operation among stakeholder is essential

We prefer using this model with those methods in order to achieve the goal of our project in forward progress and to manage unpredictable changes that will happen at any time in the future in a simple and flexible way.