**Agile Development Life Cycle**

**What is a System Development Life Cycle?**

The systems development life cycle (SDLC) is a conceptual model used in project management that describes the stages involved in an information system development project, from an initial feasibility study through maintenance of the completed application.

**What is Agile?**

This systems development life cycle works on iteration, instead of other methods e.g. waterfall where you complete one area, then move onto the next. Agile works by constantly allowing the stages to keep been explored and rejigged, for example if the design needs changing it can go back to that section and be redone.

Agile is generally up to the project of how it will be structured e.g. feasibility and business study, research, design (wireframes), programming, testing etc. This can be changed depending on the subject.

**Advantages:**

* **Customer Satisfaction:** Project is completed quicker, but secondly the customer constantly sees progress from changes etc.
* **Improves Client/internal Communication**
* **Project can be delivered in quicker stages-** e.g. Weeks instead of months
* **Continuous attention to design and technical programming**
* **Iteration-** ensures easier changes in any element, means requirements could change all the time and the changes are welcomed.
* **Save a lot of time by not having to create pointless documentation**

**Disadvantages:**

* The project can easily get taken off track if the customer representative is not clear what final outcome that they want.
* Cost can be crazy, with the constant changes.
* In case of some software deliverables, especially the large ones, it is difficult to assess the effort required at the beginning of the software development life cycle.
* There is sometimes a lack of emphasis on necessary designing and documentation.

**When is it best to use this methodology?**

* When new changes are constantly going to be added to the project.
* With large projects, because of changes
* Having options gives the ability to leave important decisions until more or better data or even entire hosting programs are available; meaning the project can continue to move forward without fear of reaching a sudden standstill.
* To implement a new feature the developers need to lose only the work of a few days, or even only hours, to roll back and implement it.

**Example Methodologies for this Life Cycle:**

DSDM, RAD, SCRUM