**Unit 22 System analysis and design – Task 1**

**Software development models**

A software development tool is a computer program that software developers use to maintain, create, and debug.

**Tool types e.g.:**

* Program analysis tools
* Testing tools
* Debugging tools
* Documentation tools
* Planning tools
* Editing tools

Software developers use a tool in a computer program which is called the software development tool.

**SDLC / Software Development life cycle**

Hardware is any physical component, tool, or machinery. Examples of hardware are keyboard, mouse, monitor etc. Software is any program built into a computer or smart device. Examples of software are Linux, Microsoft windows, PowerPoint, word etc.

Development processes take place through hardware and software. These are done through multiple steps. SDLC is used to define these steps. SDLC also shows steps involved in the development of a system.

Numerous SDLC methodologies have been developed to guide the processes involved. These include the original SDLC method and the waterfall model. Agile software development Is the name given to other SDLC methods.

Basic Stages in SDLC

* Planning and requirement analysis
* Design
* Development
* Testing
* Deployment

**Research Task - SDLC**

**1) What is SDLC?**

SDLC stands for Software Development Life Cycle. Every hardware or software system will go through a development process which can be seen as an interactive process which includes many different steps. The reason for using SDLC is to create a framework to define the steps involved in the development of a system.

Many of the SDLC methodologies have been made in order to guide the processes involved, which includes the original SDLC method, other SDLC models which today we call agile software development.

**2) Why do we need to use SDLC when creating a system?**

We use SDLC Methodologies to make software by following the project requirements. These are used to structure and plan the requirements of the system in software development.

**3) Why is it important?**

It is important weather you target hardware or software or even both. SDLC is important because it can break down the life cycle of software development. When it comes to Evaluating each part of the development is a lot easier and it helps programmers work on each stage.

**4) What different types of models are there:**

* Waterfall
* RAD
* AGILE METHODILOGIES
* PROTOTYPING

**5) What are the basic stages of SDLC? EXPLAIN EACH**

1 – Planning and Requirement Analysis

2 – Designing

3 – Development

4 – Testing

5 - deployment

REQUIREMENT phases

* The beginning phase is the requirement phase where all specifications of the project get analyses in a requirement file. As well as that, a feasible analysis also takes place in order to check if these needs are valid and reached. A main factor we must stay cautious about is the following (e.g., completing our task in our given time, spending our budget constraints) which can affect the development process. After thorough analysis, a requirement understanding document (RUD) is created.

Feasibly analysis = what you need. EG (Money, Time)

Design phase

After the requirement phase, at this phase, system and hardware specifications are prepared. For example: programming languages, user Interface, network infrastructure and many others. These help us to define the final system architecture.

Implementation/Development phase

After the design phase, the development / implementation phase begins. During this phase, source code is written. Units are then created. units are when the system is developed to a smaller program, these units are then integrated. A few of the times, the unit may get tested, this test takes place before integration.

Testing phase

At this phase, the testing team is in control of the code. The job of these testers is to check for defects in the program. In the testing phase the client is also involved, to ensure all requirements are met. If any bugs are found, then it will be fixed.

Debugging / Verification phase

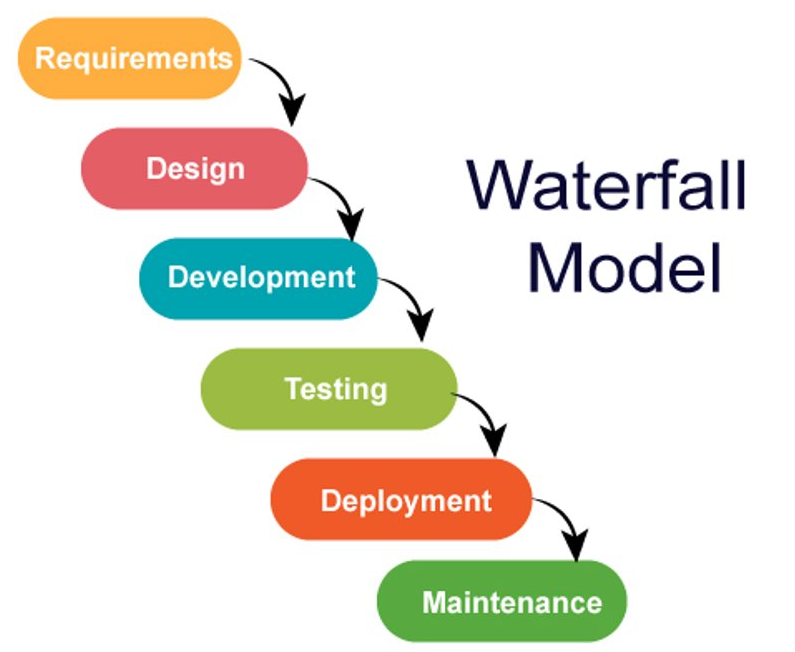
At the debugging phase, the software is released to a live environment so it can also be tested by users and clients. Once the software is deployed, it becomes available to end users. Sometimes, this phase also includes training of real time users to communicate benefits of the system.

Maintenance phase

At maintenance phase, the next objective is to support the software. This is done in order to make sure that the software runs well. If the client and users came across errors /defects /bugs, fixing them would be the main purpose of this stage.

Waterfall model

As I mentioned earlies, The waterfall model is a model used in the system development life cycle. It is used to make a system with a simple approach. The reason it is called waterfall is because the model develops from one phase to the other in a down pattern. Each phase must be completed in order to move to the next phase. Then the next phase can start and there is no overlapping in the phases.



When can the waterfall model be used?

We can use this model when we want, however it is seen to be most successful when we have a project with a big task. Projects which only demand a small number of changes should not be carried out following the waterfall model, The reason for this is because this model lacks flexibility. Whereas, if the project had a strict deadline, the waterfall model would be suitable for this in order to complete the project.

Advantages

* Easy to plan for the manager.
* It is the go-to approach for small projects.
* The entire process is well-documented and well scripted.
* Each phase is separate and complete within a given timeframe; there is less need for reworking.

Disadvantages

* Once started, changes cannot be made
* Does not work well if you have a lot of frequent changes.
* Once you agree you cannot make changes.
* It takes a huge amount of risk.
* It can lead to late detections.

**Research task – Waterfall Model**

**1) What is the waterfall model?**

The waterfall model is a model which is frequently used in the system development life cycle. It is used to make a system with a simple approach. The reason it is called waterfall is because the model moves from one phase to the other in a down pattern. Each phase must be completed in order to move to the next phase. Then the next phase can start and there is no overlapping in the phases.

**2) What phases are there? Explain each one in detail.**

REQUIREMENT phases

* During this phase, all requirements of the project get analyses to a document and feasibility analysis is done to check if these requirements are valid. It is important that we consider any limitations and constraints (for example the time we have and the budget we are working on) which can affect the development process. A RUD is then created.

Design phase

* In the design phase, our system design is prepared which specifies the hardware and system requirements, For example the data layers, programming languages, network infrastructure, user interface and many more. This helps us to define the overall system architecture.

Implementation/Development phase

* At this phase, the source code is written as per requirements are turned into a working code. The system is developed in smaller programs called units, after which these units are integrated. Sometimes, functionality of each unit is tested before integration, which is called Unit Testing.

Testing phase

* The code is then handed over to the testing team. Testers check the program for all defects, by running test cases either manually or by automation. The client is involved in the testing phase as well. To ensure all requirements are met. “All flaws and bugs detected during this phase are fixed to ensure quality assurance.” Debugging / Verification phase
* In this phase, the software is deployed into a live environment (client’s server) to test its performance. Once the software is deployed, it becomes available to end users. Sometimes, this phase also includes training of real time users to communicate benefits of the system.

Maintenance phase

* After the deployment phase, the next step is to provide support and maintenance for the software, making sure it runs smoothly. If the client and users came across errors /defects /bugs during use, fixing them is the main purpose of this stage.

**3) When can we use the waterfall model?**

When can the waterfall model be used?

This model is appropriate for projects with defined goals and requirements. Projects that demand frequent modifications should not be carried out following this approach, as this model lacks flexibility. However, if the project has strict deadlines, the waterfall model enables completion of the project, adhering to the time constraints, provided that enough resources are available.

4) What are the advantages?

* The requirements, phases and activities are well-defined. Therefore, it is easy for the project managers to plan, schedule, make use of resources and define goals. Each phase has a deliverable, thus making the entire process simple and easy to understand.
* It is the go-to approach for small projects.
* The entire process is well-documented and well scripted.
* Each phase is separate and complete within a given timeframe; there is less need for reworking.

**5) What are the disadvantages?**

Disadvantages

* Cannot be rendered to.
* Suitable for big projects.
* No more feedback after agreement.
* It includes taking risks.
* Testing starts late so you may run behind.

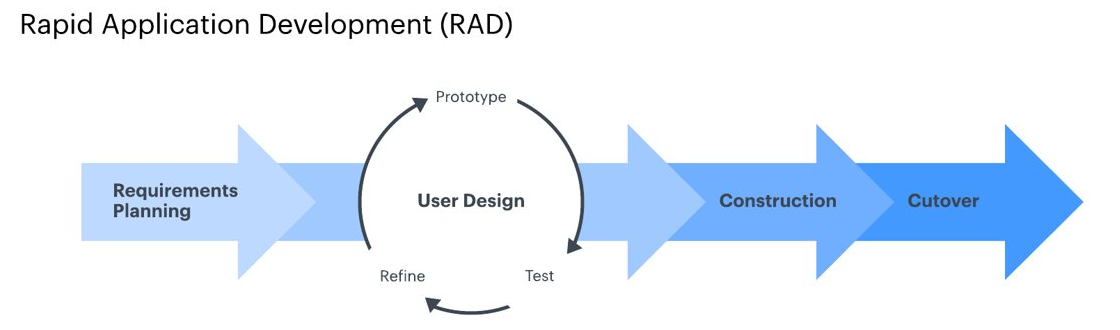
**AGILE**

AGILE METHODOLOGIES

* RAD / Rapid Application Development
* Scrum
* DSDM / Dynamic Systems Development Method
* ASD / Adaptive Software Development

**Rapid Application Development / (RAD)**

Rapid application development (RAD) is an alternative software development model which is different to the waterfall model mentioned previously, instead focuses on consistently paced delivery at high speeds. The development method has since become one of the more notably popular models on the market for software development due to its mass production efficiency with smaller tasks being handled by multiple teams at once. Generally, RAD follows a simple method of flow that includes four steps to reach its goal. This is like the waterfall model.



**Phase 1: planning**

During this phase, stakeholders will gather and finalize project requirements such as timelines, budgets, and goals. You can then seek management approval once you have defined each aspect of the project’s requirements.

**Phase 2: Design**

This phase determines the overall layout. This is done by adding changes.

**Phase 3: Constructions**

During the construction phase, there are a few steps which take place. These are: testing, coding, and fixing bugs.

**Phase 4: Cutover**

The use of the cutover phase is to test the development which you have made so for in the project. The cutover phase also applies to any further designs.

**Advantages**

* Desings and content are delivered clearly and fast. This is so all user requirements are met.
* Creates smaller projects at a faster rate.
* You can successfully measure accurate progress.

**Disadvantages**

* Highly skilled people are required.
* It is not compatible with all applications.
* Does not assist large projects.
* Not recommended when the risk is high.

**Research task – RAD model**

What is the RAD model?

Rapid application development (RAD) is an alternative software development model which is different to the waterfall model mentioned previously, instead focuses on consistently paced delivery at high speeds. The development method has since become one of the more notably popular models on the market for software development due to its mass production efficiency with smaller tasks being handled by multiple teams at once. Generally, RAD follows a simple method of flow that includes four steps to reach its goal. This is like the waterfall model.

**What phases are there? Explain each one in detail.**

Phase 1: planning

At the start, stakeholders will gather and finalize project requirements such as timelines, budgets, and goals. You can then seek management approval once you have defined each aspect of the project’s requirements.

Phase 2: Design

This phase determines the overall layout. This is done by adding changes.

Phase 3: Constructions

During the construction phase, there are a few steps which take place. These are: testing, coding, and fixing bugs.

Phase 4: Cutover

The use of the cutover phase is to test the development which you have made so for in the project. The cutover phase also applies to any further designs.

**When can we use the RAD model?**

You should use the RAD when you need to create a system. It should also be done if there Is a wide number of designers available. It will not be successful if there are no available designers. You can also use RAD when modeling and the budget is high enough to cover all costs.

**What are the advantages:**

**Advantages of the RAD model:**

* Reduced development time.
* Increases reusability of components
* Quick initial reviews occur
* Encourages customer feedback

**What are the disadvantages:**

**Disadvantages of the RAD model:**

* Depends on the teams and individual performances for identifying requirements.
* Does not work with all systems
* Requires highly skilled developers/designers.
* High dependency on modeling skills
* Quite expensive

**Scrum**

Scrum is a type of agile framework used for developing, delivering Aswell as sustaining complex products.

Scrum is made for people who are working as a team. It is recommended 10 or less members. This is done so they can break down the task into smaller goals. These goals are completed via a timeboxed iteration which is labelled as sprints. It is most common at 2 weeks but never longer than a month. The team who are working on the scrum track progress in 15 minutes from daily meetings. By the end of the sprint, the team gather reviews to demonstrate the work they have done. They can also use these reviews for continuous improvements.

**Scrum events**

Scrum is split into 5 events

* Sprint
* Sprint planning meeting
* Daily scrum
* Sprint review
* Sprint retrospective

Sprints – this region of time contains meeting and planning's. During the sprint there cannot be any changes to the goals and development teams should remain with the same members.

Sprint planning meeting – This meeting is for planning the work that must be performed during the sprint. Most of these meetings are usually 8 hours a month. However, these meetings are split into 2 parts. The beginning determines the goals of the sprint whereas the end of the meeting determines how the goals will be accomplished.

Daily Scrum – Daily scrum is a meeting which takes place each day. You may think that is a lot, but it is only 15 minutes. The reason for this is so the team can plan out what they are going to work on the next day. During the daily scrum, the team take place in reviewing what they did yesterday. Aswell as plan what goals they can achieve within the next day. There is a scum master, and his role is to make sure that the daily scrum takes place and only lasts 15 minutes.

Sprint review – After the sprint, the review takes place in order to evaluate the work accomplished during the sprint. They can then decide what the next steps are. This event takes place over 4 hours. In this time the development team can communicate with one another on the highlighted areas of the sprint.

Sprint retrospective – This event takes place over 3 hours, in contrast to the others. In this event, the entire scrum team can reflect on the sprint and decide for any improvements. This improvement can take place next spring.

**Advantages**:

* Helps teams quickly complete tasks.
* Big goals are cut down to smaller tasks to make it more manageable.
* Great for fast moving development projects.
* The team gets clear visibility.
* Short sprints enable changes.

**Disadvantages**:

* Scrum often leads to scope creep, due to lack of definite end date.
* The chances of project failures are high if individuals are not very committed.
* Needs experienced people.
* Daily meetings
* If anyone leaves during the project, it can have a significant impact.

**Research task – SCRUM**

**What is SCRUM?**

Scrum is a type of agile framework used for developing, delivering Aswell as sustaining complex products.

Scrum is made for people who are working as a team. It is recommended 10 or less members. This is done so they can break down the task into smaller goals. These goals are completed via a timeboxed iteration which is labelled as sprints. It is most common at 2 weeks but never longer than a month. The team who are working on the scrum track progress in 15 minutes from daily meetings. By the end of the sprint, the team gather reviews to demonstrate the work they have done. They can also use these reviews for continuous improvements.

**What are sprints? And why are they important?**

Sprints – this region of time contains meeting and planning's. During the sprint there cannot be any changes to the goals and development teams should remain with the same members.

**What is a daily scrum?**

Daily scrum is a meeting which takes place each day. You may think that is a lot, but it is only 15 minutes. The reason for this is so the team can plan out what they are going to work on the next day. During the daily scrum, the team take place in reviewing what they did yesterday. Aswell as plan what goals they can achieve within the next day. There is a scum master, and his role is to make sure that the daily scrum takes place and only lasts 15 minutes.

**What is a sprint review?**

Sprint review – After the sprint, the review takes place in order to evaluate the work accomplished during the sprint. They can then decide what the next steps are. This event takes place over 4 hours. In this time the development team can communicate with one another on the highlighted areas of the sprint.

**What different events are there? Explain them**

**Scrum events**

Scrum is split into 5 events

* Sprint
* Sprint planning meeting
* Daily scrum
* Sprint review
* Sprint retrospective

Sprints – this region of time contains meeting and planning's. During the sprint there cannot be any changes to the goals and development teams should remain with the same members.

Sprint planning meeting – This meeting is for planning the work that must be performed during the sprint. Most of these meetings are usually 8 hours a month. However, these meetings are split into 2 parts. The beginning determines the goals of the sprint whereas the end of the meeting determines how the goals will be accomplished.

Daily Scrum – Daily scrum is a meeting which takes place each day. You may think that is a lot, but it is only 15 minutes. The reason for this is so the team can plan out what they are going to work on the next day. During the daily scrum, the team take place in reviewing what they did yesterday. Aswell as plan what goals they can achieve within the next day. There is a scum master, and his role is to make sure that the daily scrum takes place and only lasts 15 minutes.

Sprint review – After the sprint, the review takes place in order to evaluate the work accomplished during the sprint. They can then decide what the next steps are. This event takes place over 4 hours. In this time the development team can communicate with one another on the highlighted areas of the sprint.

Sprint retrospective – This event takes place over 3 hours, in contrast to the others. In this event, the entire scrum team can reflect on the sprint and decide for any improvements. This improvement can take place next spring.

**When can we use SCRUM?**

You should use scrum for work which has a big milestone or a big release. Here is a list of recommended times for using scrum.

* When requirements are not clearly defined.
* High Development.
* Testing the solution.
* When the project owner is busy.
* When the team has self-management skills.
* When the contract is managed timely.

**Advantages and disadvantages**:

**Advantages**:

* Helps teams quickly complete tasks.
* Big goals are cut down to smaller tasks to make it more manageable.
* Great for fast moving development projects.
* The team gets clear visibility.
* Short sprints enable changes.

**Disadvantages**:

* Scrum often leads to scope creep, due to lack of definite end date.
* The chances of project failures are high if individuals are not very committed.
* Take people who have experience.
* Daily meetings.
* If anyone leaves during the project, it can have an enormous impact.

**DSDM**

DSDM stands for Dynamic Software Development Method. DSDM Is a type of RAD method that approaches to address software development issues. These can happen if you manage to miss your targeted deadline or by overspending your budget. As a member of the AGILE METHOD, DSDM goal is to complete your project on time and on your set budget. If any requirements are unclear, then the DSDM should work it out.

Main phases of DSDM:

* Pre project phase
* Lifecycle phase
* Post project phase

**Pre project phase**

This is the beginning of each project. The business problems are identified at this phase. The management decides the team to handle the project. The reason for having the pre project phase is to endure the commitment part of the project and create a start.

**Lifecycle phase**

DSDM has 4 main phases which are:

* Study – are the business requirements valid
* Functional model iteration - building a prototype to check quality
* Design and build integration - all functionalities make a fully operational system
* Implementation - product can be ready and deployed

**Post project phase**

**Advantages**

* User involvement is higher
* Basic functionalities are delivered faster
* Projects are delivered on time
* Provides access by developers

**Disadvantages**

* Not suitable for small projects
* As it is newer than old models it is not as common
* DSDM can be restrictive and hard to work

**Research task – DSDM**

**What is DSDM**

DSDM stands for Dynamic Software Development Method. DSDM Is a type of RAD method that approaches to address software development issues. These can happen if you manage to miss your targeted deadline or by overspending your budget. As a member of the AGILE METHOD, DSDM goal is to complete your project on time and on your set budget. If any requirements are unclear, then the DSDM should work it out.

**What are the phases? Explain each phase?**

Main phases of DSDM:

* Pre project phase
* Lifecycle phase
* Post project phase

**Pre project phase**

This is the beginning of each project. The business problems are identified at this phase. The management decides the team to handle the project. The reason for having the pre project phase is to endure the commitment part of the project and create a start.

**Lifecycle phase**

DSDM has 4 main phases which are:

* Study – are the business requirements valid
* Functional model iteration - building a prototype to check quality
* Design and build integration - all functionalities make a fully operational system
* Implementation - product can be ready and deployed

**Post project phase**

* Post project phase is the last phase of DSDM. Just like the implementation task, the final building is viewed by the team to ensure that it meets the business expectations. This phase also consists of further improvements/maintenance of the product, bugs are fixed.

**What are advantages and disadvantages**

**Advantages**

* User involvement is higher
* Basic functionalities are delivered faster
* Projects are delivered on time
* Provides access by developers

**Disadvantages**

* Not suitable for small projects
* As it is newer than old models it is not as common
* DSDM can be restrictive and hard to work

**ASD**

ASD stands for Adaptive software development. ASD is a straight outgrowth of an early framework. The reason for ASD is to aim to enable teams to adapt quickly and effectively to changing requirements. The ASD approach encourages groups to develop according to a 3 step / phase process: speculate, collaborate, and learn.

It is key to know that these 3 will not overlap. Therefore, there is no feasible way to do them at the same time.

**Speculate**

In Adaptive software development, the given term planning is swapped by the term speculate. When speculating, the term does link to planning. However, it acknowledges the uncertainty of the problem.

**Collaborate**

To collaborate, you will need the ability to work jointly to gain results. You also need to be able to make choices Aswell as share your knowledge.

**Learn**

After every iteration, a review should take place. This must be done by both customers and developers. By having this review, it can tell the team about changes in product. The team are most likely to learn of smaller tasks than larger therefor the iteration must be kept short to help the team.

**Advantages**

* Involved in the final users
* Succeed in early deliveries
* Creates more transparency between the two (client and developers)

**Disadvantages**

* Creates testing for each stage
* Scope creep can be created by the lack of iteration
* It is difficult to facilitate the involvement of users

**Research task – ASD**

**What is ASD**

ASD stands for Adaptive software development. ASD is a straight outgrowth of an early framework. The reason for ASD is to aim to enable teams to adapt quickly and effectively to changing requirements. The ASD approach encourages groups to develop according to a 3 step / phase process: speculate, collaborate, and learn.

It is key to know that these 3 will not overlap. Therefore, there is no viable way to do them at the same time.

**What stages are there? Explain each?**

**Speculate**

In Adaptive software development, the given term planning is swapped by the term speculate. When speculating, the term does not link to planning. However, it acknowledges the uncertainty of the problem.

**Collaborate**

To collaborate, you will need the ability to work jointly to gain results. You also need to be able to make choices Aswell as share your knowledge.

**Learn**

After every iteration, a review should take place. This must be done by both customers and developers. By having this review, it can tell the team about changes in product. The team are most likely to learn of smaller tasks than larger therefor the iteration must be kept short to help the team.

**What are the advantages and disadvantages**

**Advantages**

* Involved in the final users
* Succeed in early deliveries
* Creates more transparency between the two (client and developers)

**Disadvantages**

* Creates testing for each stage
* Scope creep can be created by the lack of iteration
* It is difficult to facilitate the involvement of users

**Comparing models**

Compare waterfall model and RAD. Evaluate your answers.

**Waterfall**

The waterfall model is a model used in system development life cycle. It is used to make a system with a simple approach. The reason it is called waterfall is because the model develops from one phase to the other in a down pattern. Each phase must be completed in order to move to the next phase. Then the next phase can start and there is no overlapping in the phases.

Advantages

* Easy to plan for the manager.
* It is the go-to approach for small projects.
* The entire process is well-documented and well scripted.
* Each phase is separate and complete within a given timeframe; there is less need for reworking.

Disadvantages

* Once started, changes cannot be made
* Does not work well if you have a lot of frequent changes.
* Once you agree you cannot make changes.
* It takes a huge amount of risk.
* It can lead to late detections.

**RAD**

Rapid application development (RAD) is another software development model which is different to the waterfall model which I have mentioned above. Instead, the RAD focuses on getting delivery at a high speed. The development method has since become one of the more notably popular models on the market for software development due to its mass production efficiency with smaller tasks being handled by multiple teams at once. Generally, RAD follows a simple method of flow that includes four steps to reach its goal. This is like the waterfall model.

**Advantages**

* Desings and content are delivered clearly and fast. This is so all user requirements are met.
* Creates smaller projects at a faster rate.
* You can successfully measure accurate progress.

**Disadvantages**

* Highly skilled people are required.
* It is not compatible with all applications.
* Does not assist large projects.
* Not recommended when the risk is high.

**Is scrum a good framework to use? WHY?**

Scrum is a type of agile framework which we use for developing, Aswell as sustaining complex products.

Scrum is dedicated to people who are working as a team. It is recommended to have 10 or less members. This is done so they can break down the task into smaller goals. These goals are completed via a timeboxed iteration which is labelled as sprints. It is most common at 2 weeks but never longer than a month. The team who are working on the scrum track progress in 15 minutes from daily meetings. By the end of the sprint, the team gather reviews to demonstrate the work they have done. They can also use these reviews for continuous improvements.

**Advantages**:

* Helps teams quickly complete tasks.
* Big goals are cut down to smaller tasks to make it more manageable.
* Great for fast moving development projects.
* The team gets clear visibility.
* Short sprints enable changes.

**Which is better ASD or DSDM? Evaluate your answer**

ASD stands for Adaptive software development. ASD is a straight outgrowth of an early framework. The reason for ASD is to aim to enable teams to adapt quickly and effectively to changing requirements. The ASD approach encourages groups to develop according to a 3 step / phase process: speculate, collaborate, and learn.

It is key to know that these 3 will not overlap. Therefore, there is no viable way to do them at the same time.

**Advantages**

* Involved in the final users
* Succeed in early deliveries
* Creates more transparency between the two (client and developers)

**Disadvantages**

* Creates testing for each stage
* Scope creep can be created by the lack of iteration
* It is difficult to facilitate the involvement of users

**DSDM**

DSDM stands for Dynamic Software Development Method. DSDM Is a type of RAD method that approaches to address software development issues. These can happen if you manage to miss your targeted deadline or by overspending your budget. As a member of the AGILE METHOD, DSDM goal is to complete your project on time and on your set budget. If any requirements are unclear, then the DSDM should work it out

**Advantages**

* User involvement is higher
* Basic functionalities are delivered faster
* Projects are delivered on time
* Provides access by developers

**Disadvantages**

* Not suitable for small projects
* As it is newer than old models it is not as common
* DSDM can be restrictive and hard to work

**Compare them all to each other**

**Waterfall**

The waterfall model is a model used in system development life cycle. It is used to make a system with a simple approach. The reason it is called waterfall is because the model develops from one phase to the other in a down pattern. Each phase must be completed in order to move to the next phase. Then the next phase can start and there is no overlapping in the phases.

**Advantages**

* Easy to plan for the manager.
* It is the go-to approach for small projects.
* The entire process is well-documented and well scripted.
* Each phase is separate and complete within a given timeframe; there is less need for reworking.

**Disadvantages**

* Once started, changes cannot be made
* Does not work well if you have a lot of frequent changes.
* Once you agree you cannot make changes.
* It takes a huge amount of risk.
* It can lead to late detections.

**RAD**

Rapid application development (RAD) is another software development model which is different to the waterfall model which I have mentioned above. Instead, the RAD focuses on getting delivery at a high speed. The development method has since become one of the more notably popular models on the market for software development due to its mass production efficiency with smaller tasks being handled by multiple teams at once. Generally, RAD follows a simple method of flow that includes four steps to reach its goal. This is like the waterfall model.

**Advantages**

* Desings and content are delivered clearly and fast. This is so all user requirements are met.
* Creates smaller projects at a faster rate.
* You can successfully measure accurate progress.

**Disadvantages**

* Highly skilled people are required.
* It is not compatible with all applications.
* Does not assist large projects.
* Not recommended when the risk is high.

**Scrum**

Scrum is a type of agile framework which we use for developing, Aswell as sustaining complex products.

Scrum is dedicated to people who are working as a team. It is recommended to have 10 or less members. This is done so they can break down the task into smaller goals. These goals are completed via a timeboxed iteration which is labelled as sprints. It is most common at 2 weeks but never longer than a month. The team who are working on the scrum track progress in 15 minutes from daily meetings. By the end of the sprint, the team gather reviews to demonstrate the work they have done. They can also use these reviews for continuous improvements.

**Advantages**:

* Helps teams quickly complete tasks.
* Big goals are cut down to smaller tasks to make it more manageable.
* Great for fast moving development projects.
* The team gets clear visibility.
* Short sprints enable changes.

**ASD**

ASD stands for Adaptive software development. ASD is a straight outgrowth of an early framework. The reason for ASD is to aim to enable teams to adapt quickly and effectively to changing requirements. The ASD approach encourages groups to develop according to a 3 step / phase process: speculate, collaborate, and learn.

It is key to know that these 3 will not overlap. Therefore, there is no viable way to do them at the same time.

**Advantages**

* Involved in the final users
* Succeed in early deliveries
* Creates more transparency between the two (client and developers)

**Disadvantages**

* Creates testing for each stage
* Scope creep can be created by the lack of iteration
* It is difficult to facilitate the involvement of users

**DSDM**

DSDM stands for Dynamic Software Development Method. DSDM Is a type of RAD method that approaches to address software development issues. These can happen if you manage to miss your targeted deadline or by overspending your budget. As a member of the AGILE METHOD, DSDM goal is to complete your project on time and on your set budget. If any requirements are unclear, then the DSDM should work it out

**Advantages**

* User involvement is higher
* Basic functionalities are delivered faster
* Projects are delivered on time
* Provides access by developers

**Disadvantages**

* Not suitable for small projects
* As it is newer than old models it is not as common
* DSDM can be restrictive and hard to work