ASSIGNMENT 01

Name : Laveena Rajendran

Year : Year III semester I

Department : Department of Computing &

Information systems

Subject : Rapid Application Development

Date : 31st March 2020

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Content** | **Page No** |
| 01. | Introduction   1. What is RAD? 2. Best practices in RAD | 02 |
| 02. | Joint Application development | 03 |
| 03. | Principled Negotiation | 03 |
| 04. | Requirements Scrubbing | 03 |
| 05. | Timebox Development | 04 |
| 06. | Reuse | 04 |
| 07. | Goal Setting | 04 |
| 08. | Outsourcing | 05 |
| 09. | Measurement | 05 |
| 10. | Rapid Development Languages & Tools | 05 |
| 11. | Daily Build & Smoke Test | 06 |
| 12. | Inspections | 06 |
| 13. | Evolutionary Delivery | 07 |
| 14. | Evolutionary Prototyping | 07 |
| 15. | Throwaway Prototyping | 08 |
| 16. | User Interface Prototyping | 08 |
| 17. | References | 09 |

**INTRODUCATION**

1. **What is RAD?**

Rapid Application Development is a software development methodology that is an iterative approach which is used in building software applications by rapid prototyping. It solves the issues of restrictions of waterfall model, therefore RAD is used as an alternative to the waterfall model. RAD cannot be used for every situation and it provides competitive advantages to combat thee rivals such as

* Progress is measurable and development time is reduced.
* Rapid code generation
* Easily Modifiable
* User feedback is faster and constant
* Early system integration

1. **Best Practices in RAD**

Rapid Application Development uses predefined techniques and tools to produce software application which are simply known as ‘Vest Practices’ speed up or helps in RAD.

**JOINT APPLICATION DEVELOPMENT**

It is a kind of requirements definition and user interface design that involves intensive off meeting with the end user, executives, developers or the client in the initial stages of the software application such as design and development through a successful collaborative workshops known as JAD sessions.

In comparison with other traditional practices, JAD approach leads to the faster development with great client satisfaction as the client or the end user involved from start to end of the software application development process. It reduces the no of costly downstream requirements changes.

It is easy for a developer to investigate on the system requirement, by interacting with the client through interviews continuously, so the final product is created as soon as possible.

**PRINCIPLED NEGOTIATION**

It is a most useful strategy which is used in stages like requirement analysis, schedule creation, feature change discussions. It finds a deal which benefits all parties involved in a software application development. If the parties able to achieve a good relationship, then the objective becomes common interest which can help to generate options for mutual gain

Fundamental practices followed are

* Separate the people from problem
* Focus on interest and notion positions
* Invent options for a mutual gain
* Insist on objective criteria

Main benefits is it clarifies the expectations and identify exactly what is needed to carry out for a successful project. By this it reduces the time taken to identify and understand the requirements needed, and the output is quickly produced.

**REQUIREMENTS SCRUBBING**

Here the product specification is analysed and examined carefully. From that the unnecessary and overly complexed requirements, features are omitted. Therefore the size of the product is reduced. By this the cost of that particular product is decreased so as the time taken for the development process, and it is received quickly by the end user.

**TIMEBOX DEVELOPMENT**

It is simply known as construction time practice which helps the development team to organize the performance of work and manage scope, with the sense of urgency. By this, the most important feature of the project is highly focused.

Using this practice, the product/ outcome is quickly as it redefine the product to fit the schedule instead of redefining the schedule to fit the project. Some of the advantages of Timebox development practices are

* Prioritize tasks
* Track the progress
* Communication is increased
* Record a history pf completed work
* Increase the employee engagement

**REUSE**

This is the generally used long term strategy used by almost every organization. Reuse provides the greater schedule management and effort savings than the other best practices used in RAD. Because it allows the novice developer to use existing component to assemble new programs rather than creating the old ones again and again.

Some of the advantages of reuse are

* Increase the software productivity
* Shorten the software development time
* Improved software system interoperability
* Software is developed with the fewer people
* Reduced cost

**GOAL SETTING**

During this practice, the client or the project manager explains clearly about their requirements that is to be built in the final product. Developers work hard to achieve their goal within a shorter development time as goal setting contributes to the motivation.

Common problem in a software development process like undesired to define the objectives and commit to them for a complete project is solved by this practice. Thereby it helps to develop the project on time without any complexity in identifying the software components.

**OUTSOURCING**

The name itself implies that the sources are obtained from outside the organization instead of developing it in house. Here the third party software contractor develop the particular software component and provide it to the particular company where sometimes the charges for the development of that specific component is less than producing it in own organization.

By this the time taken to develop the whole product is less and even with the cos benefit, because the team members needed to develop the whole product is dramatically reduced.

Advantages of outsourcing towards rapid application development

* Focused strategy
* Flexibility
* Improved compliance
* Reduced time to market
* Technological advances

**MEASUREMENT**

It is one of the RAD best practice which provide short term motivational benefits along with long term quality, cost and schedule benefits. It is implemented to individual project by project team or individual team members. Basically it provides the solutions to the problems like poor estimations, poor scheduling and poor progress visibility.

By this it increases the project development process and the outcome is delivered as soon as possible. Measurement can be used be used and benefited by any organization or project at some level.

For a successful output, the measurement should be highly managed and be carried out through a permanent management group.

**RAPID DEVELOPMENT LANGUAGES & TOOLS**

It is nothing but any programming language which offers faster implementation than traditional third generation languages. It reduces the development time by decreasing the amount of construction that needed for building the product and shorten the construction cycle where these shorter cycles make incremental lifecycles like evolutionary prototyping practical.

**DAILY BUILD & SMOKE TEST**

It is a construction stage process where software application is built completely every day and verified its basic operations by a series of tests. It can be carried out even at the middle of the software process.

Daily build and smoke test reduces the probability of time consuming risks which can be happened commonly such as poor quality, unsuccessful integration, and poor progress visibility. This can be used in project with any size and complexity in an effective manner.

Advantages of daily build and smoke test are

* Supports easier defect diagnosis
* Improves morale
* Checks for broken builds
* Add revisions to the build only when it makes sense to do so
* Establish a build group

**INSPECTIONS**

This is the process of finding errors and deficiencies in a developed software product and ensure that it fulfils the client’s requirement. Inspections are effective at finding errors than the execution testing.

During software inspection, all participants who are well trained in review practices have set a role. These roles are played during the review meeting which helps to identify the additional errors.

As these inspections help to find out the errors at early stages, it benefits in avoiding cost, downstream rework. Therefore the rapid development of the software project is achieved.

Process during inspections

Planning 🡪 Overview Planning 🡪 Preparation 🡪

Inspection Meeting 🡪 Rework 🡪 Follow up

Some of the advantages are

* Finds defects or fault at early
* Improve coding style, best practices
* Educates and trains developers
* Improved communications

**EVOLUTIONARY DELIVERY**

It is a life cycle model which is a combination of incremental and iterative model of software development life cycle. The estimated schedule is achieved by the involvement of users as they provide feedback on product for the planning stage of the next cycle, from this feedback, the developers get the necessary information and requirement and change, plan, process according to the client’s need.

Also, without any necessary need, it won’t deliver the final product any faster. Advantages of evolutionary delivery are

* Reduce the risks involved in the software product
* Minimizes the overall cost
* Reduces the problems that arise during testing process

**EVOLUTIONARY PROTOTYPING**

It is also a life cycle where the development team creates the prototype first. Then this prototype is checked and analysed by the client and the feedback is received. This starts by prototyping the user interface and then evolve to the complete system from it.

For a successful evolutionary prototyping includes experienced developers, good schedule and budget management and also the managing the activities involved in prototyping. This process continues by constructing subsequent prototypes each includes details of added functionalities and improvements on previous software components until the final product is produced.

Benefits of evolutionary prototyping

* Effort to construct prototype is not wasted
* Faster when compared to waterfall model
* User is involved from start to end of the software product
* Problems are discovered early therefore the risks are reduced
* Can be used for projects with unstable requirements

**THROWAWAY PROTOTYPING**

It is a faster practical implementation of the system that uses programming languages or development practices or both which are faster than the target practices and languages and used only to explore ideas, and to obtain the feedback from the user.

It helps to identify and understand the requirements problems and thrown away. These type of prototype involves in creating a working model of various system component at early stage.

Advantages of Throwaway prototyping

* Allow the customer to further define the product as it is being developed.
* Prototype is not documented and it is not intended to be part of the final product
* It can be written in a different language that is not used in the final system to speed up the process of prototype creation.
* Reduce the risks in project
* Has short project timeline

**USER INTERFACE PROTOTYPING**

Here the user interface is developed first to identify the system’s further requirements and the necessary user interface design improvements. These kind of prototypes are not necessarily included in the final product it can be discarded if it is not needed.

This choice of discarding to keeping the prototype leads to the success. As the user interface is almost developed initially the customer changes can be carried out at quickly and the complexity and miscommunications between the client and the developer team are ignored and therefore the final product is delivered quickly.

Advantages of UI prototyping

* Easy to explain about the system to customer
* Can benefit as the problems are identified at early development
* Flexibility
* Relationship among development team and the client are improved

**REFERENCES**

Lecture notes

[www.answerspoint.com](http://www.answerspoint.com)

[www.coursehero.com](http://www.coursehero.com)

graphicmint.com

innolution.com

stevemccpnnell.com