Unit : 8 Risk Management

* Difference Between Reactive and Proactive Risk strategies.

=>

	Reactive	Proactive
1	In this approach, we have to use Don't worry approach.	In this approach we have to avoid the Risk.
2	There is no plan create for the Risk until is identify.	There is plan create for the avoid Risk.
3	In this approach, Risk is moniter in lightly way.	In this approach Risk is identify with its probability.
4	In this method, we can use the any type of risk method.	In this method, we have to always try to do avoid the Risk

5	Risk management is done after the Risk is accrues.	Risk management is done before the Risk is accrues.

* Explain Risk Identification Checklist in Software.

=> Using Risk Identification we can identify the Risk in the Software.

This are the Types of Risk which is accures in the software.

(i) Product Size Risks :

- Estimated size of Lines of codes
- Estimated size of Program
- Estimate the number of Program
- Estimate the size of database which is use in software.
- Estimate the number of users

(ii) Business Impact Risks :

- Numbers of Customers
- Consistency of Software
- Delivery Deadline
- Number of System which is use in software.
- Governmental Constraints

~~(iii)~~

(iii) Customer Related Risk :

- Customer Idea for the software
- Customer Time for formal communication
- Customer Participation in Software Reviews.
- Customer Behaviour.

(iv) Process Risk :

- Senior Management Support
- Team member Support
- Formal Technical Review.
- Mechanism Control the Changes Requirement of the software.
- Tracking.

(v) Technical Issues :

- Specification Techniques used in software.
- Specific method to analysis of software.
- Data and Architectural Design
- Code Documentation
- Test case

(vi) Technical Risks :

- Technology which is use in software.
- Customer's Requirement
- Software Interface
- Specialized User Interface
- Constraints Performance
- Customer's uncertain requirement.

(vii) Development Environment :

- Software management tools.
- Tools for Analysis and Design
- Code Generators
- Software Configuration Managment.

* Explain Different Software Risk with its Mitigation, Monitoring and management.

=> This are the different types of Software Risk.

(i) Computer ~~Risk~~ Crash Risk :

- Mitigation - Computer Crash is not a crucial problem, but Computer Crash can destroy or loss the all the data of Software.

- Monitoring - Developer have to monitor the working development environment when Software is develope.

- Management : When we try to develope the software then we have to always take backup of the data.

(ii) Late Delivery :

- Mitigation - Late Delivery is a critical problem because it is increase the cost and resource management in Software.

- Monitoring - For reduce the late delivery time, we have to always monitore and Track the software development.

- Management - For avoid the Late Delivery, we have to request to Customer to extend the Software Delivery Deadline and always work with backup or extra days.

(iii) Changes in Requirements :

- Mitigation : At the time of collect the requirement, we have to insure that product we will producing it is equivalent to customer.

- Monitoring : We have to always monitor the all the requirement which is give by the customer.

- Management : At the time of creation, we have to take customer review so, customer is clear about the software requirements.

(iv) Database is not stable :

- Mitigation : Developer have to contact with the database or its function which is interact ~~wt~~ with the database.

- Monitoring : We have to monitore the database error or database connect error problem.

- Management : We have to manage the all the database in the software which is use to transfer or interact with data in the software.

(v) End Users Resist System :

- Mitigation : Software User - interface design in a way that can provide convenient user interface.

- Monitoring : We have to always monitor the end user interface which is provides batter performance to the end user.

- Management : At the time of coding, we have to create a code in a such a way that can provides batter performance.