Lecture # 21

Prototyping

Prototyping

- It is the technique of constructing a partial implementation of a system so that customers, users, or developers can learn more about a problem or a solution to that problem

Prototype - 1

- An initial version of the system under development, which is available early in the development process
- A prototype can be a subset of a system, and vice versa, but they are not the same

Prototype - 2

- In hardware systems, prototypes are often developed to test and experiment with system designs
- In software systems, prototypes are more often used to help elicit and validate the system requirements. There are other uses also

Prototype - 3

- It should be easy for a prototype to be developed quickly, so that it can be used during the development process
- Prototypes are valuable for requirements elicitation because users can experiment with the system and point out its strengths and weaknesses. They have something concrete to criticize

Types of Prototyping

- Throw-away prototyping
- Evolutionary prototyping

Throw-away Prototyping - 1

- Intended to help elicit and develop the system requirements
- The requirements which should be prototyped are those which cause most difficulties to customers and which are the hardest to understand. Little documentation is needed

Throw-away Prototyping - 2

- Determine the feasibility of a requirement
- Validate that a particular function is really necessary
- Uncover missing requirements
- Determine the viability of a user interface

Throw-away Prototyping - 3

- Writing a preliminary requirements document
- Implementing the prototype based on those requirements
- Achieving user experience with prototype

Throw-away Prototyping - 4

- Writing the real SRS
- Developing the real product

Evolutionary Prototyping - 1

- Intended to deliver a workable system quickly to the customer
- The requirements which should be supported by the initial versions of this prototype are those which are well-understood and which can deliver useful end-user functionality

Evolutionary Prototyping - 2

- Documentation of the prototype is needed to build upon
- This process repeats indefinitely until the prototype system satisfies all needs and has thus evolved into the real system

Evolutionary Prototyping - 3

- Evolutionary prototype may not be built in a ‘dirty’ fashion. The evolutionary prototype evolves into the final product, and thus it must exhibit all the quality attributes of the final product

Comparison of Prototyping - 1

	Throwaway	Evolutionary
Development approach	Quick and dirty. No rigor	No sloppiness. Rigorous

Comparison of Prototyping - 2

	Throwaway	Evolutionary
What to build	Build only difficult parts	Build understood parts first. Build on solid foundation

Comparison of Prototyping - 3

	Throwaway	Evolutionary
Design drivers	Optimize development time	Optimize modifiability
Ultimate goal	Throw it away	Evolve it

Prototyping Benefits - 1

- The prototype allows users to experiment and discover what they really need to support their work
- Establishes feasibility and usefulness before high development costs are incurred

Prototyping Benefits - 2

- Essential for developing the 'look and feel' of a user interface. Helps customers in 'visualizing' their requirements
- Forces a detailed study of the requirements which reveals inconsistencies and omissions

Prototyping Costs

- Training costs
  - Prototype development may require the use of special purpose tools
- Development costs
  - Depend on the type of prototype being developed

Prototyping Problems - 1

- Extended development schedules
  - Developing a prototype may extend the schedule although the prototyping time may be recovered because rework is avoided

Prototyping Problems - 2

- Incompleteness
  - It may not be possible to prototype emergent system requirements

Additional Benefits of Prototyping

- Developing a system prototype is worth the investment in time and money
- Real needs of the customers will be reflected in the requirements set
- Rework will be reduced
- Defect prevention

Developing Prototypes

- Conventional system development techniques usually take too long, and prototypes are needed early in the elicitation process to be useful
- Rapid development approaches are used for prototype development

Approaches to Prototyping

- Paper prototyping
- 'Wizard of Oz' prototyping
- Executable prototyping

Paper Prototyping - 1

- A paper mock-up of the system is developed and used for system experiments
- This is very cheap and very effective approach to prototype development
- No executable software is needed

Paper Prototyping - 2

- Paper versions of the screens, which might be presented to the user are drawn and various usage scenarios are planned
- For interactive systems, this is very effective way to find users' reactions and the required information

‘Wizard of Oz’ Prototyping - 1

- A person simulates the responses of the system in response to some user inputs
- Relatively cheap as only user interface software needs to be developed
- The users interact through this user interface software and all requests are channeled to the a person, who simulates the system’s responses

‘Wizard of Oz’ Prototyping - 2

- This is particularly useful for new systems, which are extensions of existing software systems, and the users are familiar with the existing user interface
- The person simulating the system is called ‘Wizard of Oz’

Executable Prototyping - 1

- A fourth generation language or other rapid development environment is used to develop an executable prototype
- This is an expensive option and involves writing software to simulate the functionality of the proposed system

Executable Prototyping - 2

- 4GLs based around database systems are useful for developing prototypes, which involve information management

Executable Prototyping - 3

- Visual programming languages such as Visual Basic or ObjectWorks
- These languages are supported by powerful development environments, which include access to reusable objects and user interface development utilities. Support for database-oriented applications is not that strong

Executable Prototyping - 4

- Internet-based prototyping solutions based on WWW browsers and languages. Here, we a ready-made user interface and Java applets can be used to add functionality to the user interface

Comments on Prototyping

- Prototyping interactive applications is easier than prototyping real-time applications
- Prototyping is used better to understand and discover functional requirements, as compared to non-functional requirements

Summary

- Discussed different aspects of prototyping, its types, and how it is useful in requirements engineering, particularly requirements elicitation
- We also discussed approaches to prototyping: paper prototyping, 'Wizard of Oz' prototyping, and 'automated prototyping'