PROTOTYPING

29/11/2017

Presented by: Shuchita Singh
Outline

- Requirement Analysis Overview
- Prototyping
- Life cycle of Prototyping
- Types of prototyping techniques
- Advantages and Disadvantages
- Case Studies
- Conclusion
What is Requirement Analysis?

Source: http://www.marcocioffi.com/archives/2005/04/requirements-engineering/
What is Requirement Analysis?

- *Understanding* the *customer* to find out the functionalities of the target system and documenting them
  - Understanding + Customer = TOUGHEST TASK
  - Requirements may change over time

- Document describing **what** is to be done
  - Not **how**
What is Requirement Analysis?

- EXAMPLE:
  - IBM Deep Blue beat the world chess champion in 1997

<table>
<thead>
<tr>
<th>✓</th>
<th>✗</th>
</tr>
</thead>
<tbody>
<tr>
<td>Win a match against Human</td>
<td>Use a minimax search to decide the move</td>
</tr>
<tr>
<td>3 minutes per move</td>
<td>Heuristic function</td>
</tr>
<tr>
<td>‘Learn’ from opponent’s moves</td>
<td>Use alpha-beta algorithm</td>
</tr>
</tbody>
</table>
What is the need of Prototyping?

- For a large and complex system
  - Requirements are vast and hidden

- Analysing and understanding the requirements difficult
  - Syncing your thoughts to user’s – Main Challenge
  - Bridging the GAP

- When all system requirements are not known well ahead of time
  - Find out the hidden requirements and challenges
Bridge the GAP

Customer

Description

Interpreted

Requirement Analyst
Prototyping

- An **early approximation of a final system** or product is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved[1]

- A good candidate for Prototype is:[7]
  - ‘The user is unable to articulate (or "prespecify") his or her requirements in any form and can only determine the requirements through a process of trial and error.’
Dimensions of Prototyping

- Neilsen in his book Usability Engineering provides two dimensions of prototyping:
  - Horizontal Prototyping – broad overview of entire system
  - Vertical Prototyping – obtain detailed requirement of a particular function
Life cycle of Prototyping

Source: http://www.enterox.com/cloud/software-prototyping.htm
Types of Prototyping Techniques

- Throwaway Prototyping
- Evolutionary Prototyping
- Incremental Prototyping
Throwaway Prototyping

- Referred to as Rapid Prototyping

- “It involves creating a working model of various parts of the system at a very early stage, after a relatively short investigation”[2]

- Once goals are achieved the prototype built is ‘discarded’

- Often used for requirements identification and clarification[3]

- Example: Is it possible to store user information in database from the text file?
Evolutionary Prototyping

- Evolutionary prototype forms the heart of the new system, and the improvements and further requirements will then be built[4]
- It is structured to be robust
- Constantly refined and rebuilt and eventually become a solution
- Developer does not implement poorly understood features
- Example: How different components of a system work. Fix the discovered bugs and continue working
Incremental Prototyping

- The incremental approach is like 'building blocks'; incrementing each time a new component is added or integrated, based on an overall design solution[5]
- The final system is built when all the ‘blocks’ are put together
- Frequent user feedback on working parts
- Example: Application allows user to cut, copy and paste text but not insert images or animations
Advantages of Prototyping

- Cost Saving
  - Possible to lower the cost of development (Bohem et al. 1984)

- Produce correct system
  - Ensure that the solution does what it is supposed to do - not what the developer thinks it ought to do, or how

- Proper requirement analysis
  - Identify and address problems early in the software lifecycle

- Highly satisfied user
Advantages of Prototyping

- Bridging the gap
  - Explore ideas and exchange feedback with the client
- Useful reference point[5]
  - It can be referred back to when needed
Disadvantages of Prototyping

- User may expect the end product to be like prototype
- Insufficient analysis[4]
  - Might distract from properly analysing the project
- Large amount of time spent on building the prototype
- High expectations for productivity with insufficient effort behind the learning curve[6]
Case Studies

1. Requirements engineering by prototyping: Experiences in development of estimating system
   - by M A Stephens and P E Bates

2. Prototyping in Requirement Engineering
   - By Shuchita Singh
Case Study 1

- **OBJECTIVE**

- Metal-plating company needed a better and reliable method of pricing and estimation. Till now they relied on specialist. What they wanted was:
  - Current practices in pricing and estimating to be analyzed
  - The cause of problems reported
  - An automated system to streamline processing specified
Case Study 1

• PROBLEM
  
  • Relationships between costs and prices remained unclear
  • Inconsistency with quotations
  • Manual quotations were expensive to produce
  • No rules to discover determinants into quotations
Case Study 1

- **SOLUTION**

- Evolutionary prototyping approach for interface
  - Led to:
    - Change in system requirement
    - Need for **functional prototyping** for important components
Case Study 2 [Still Working!]

- Large scale Seeking Game using Android Phones
- How building a prototype help them in the requirement gathering phase?
- Learning curve
- What problems did they realize once the prototype was generated?
- How did they solve it?
- Was building a prototype helpful or a waste of time?
Conclusion

- What is prototyping in RE?
- What is the need of prototyping?
- When should prototyping be done?
- What is a general prototyping life cycle?
- What are the types and uses of prototyping?
- Should it be a part of software development process?
Reference