→Able to use past data to do current project.
→Can more accurately determine the effort required of job to be done.
For systematic approach we need to be disciplined.
Engineering Note Book: (Ref: Video of "Engineering Note Book")(00:21:40)
→Engineering note book
Memory aid
Manage Change
Asserting intellectual property ownership

→Typical Engineering note books
Traditional Engineering Notebook
→Notes
What work is done and what work is not done. Data used and how it is used.
→Status
It is about what happened and where are you in work. It can help your team mates know where you
→Lessons Learnt
What we learnt and what we actually want to learn. Why the gap occurred and what are you doing to
→Plans
Tool to build alignment about the work to be done and how to do it.
→Conclusion
Use ENB effectively and simply. Use it as a memory aid.
"
(Reference: video of intro to SEF)

"Test Driven Development (TDD): (Ref: Video of "Test-Driven Development") (00:21:57)
There are many ways to implement TDD. Not all are optimal and can be supported by everyone. W
more time on writing tests we can spend time on getting requirements correctly.
V- Model:
It points out each and every major aspect that is needed to be performed. Each activity has some te
disadvantages are the testing is done after the development of model fully.
Royce's model:
Every activity has a validation and test to be performed at same time. When we have some error we
happen after the code is written:
1 Code is good and test is good
2 Code is good and test is bad
3 Code is bad and test is good
4 code is bad and test is bad
Questions we must address:
1. Should I try to gather all requirements?
2. How much design and documentation is enough?

3. When I stop producing detailed design and start coding?
Risk Management:
It is a very powerful tool. If we are certain that something won't change, address it proactively. If we
system.
Powerful Tools:
Pair programming
Before design -black box test
After coding - white box test
"
(Reference: video of ENB)

"State Diagrams: (ref: video of "State Diagram", http://www.tutorialspoint.com/uml/uml_statechart_
It is one of the five UML diagrams used to model dynamic nature of system. They define different st
Systems are systems that react to external or internal events.
Purpose:
1. To model dynamic aspects of systems.
2. To model lifetime of a reactive system.
3. To state different states of object during its life time.
4 Define a state machine to model states of an object.
Before drawing state diagram we must clarify following points:
1 Identify important objects to be analysed.
2. Identify the states.
3. Identify the events.
"
(Reference: video of state diagram)

"Introduction to requirements: (ref: video of "introduction to requirements") (00:32:10)
Requirements:
It is branch of software engineering concerned with the real-world goals for, functions of, and cons
factors to precise specifications of software behaviour, and to their evolution over time and across
Requirements Elicitation:
Gathering and balancing the information needed to fully appreciate the problem.
Prioritize needs and give proper weights of the key factors is a crucial aspect of this activity.
Understanding, recording, and monitoring the requirements are crucial.
Remaining neutral and not biasing the input requires skill
Establish the business goals first
Work down from these business goals to the more technical goals to clarify things.
Stay in the problem space.
Elicitation Techniques:
Questionnaires, surveys, interviews
Group methods
Mock-ups and throwaway prototypes
Contextual inquiry
Modelling and analysis:
Modelling is a tool that allows us to identify things that are important and utilise them and separate
Two kinds of models

Descriptive models:
Describe aspects of the system that the modeller believes are useful. Helps us to compare and co
Predictive models:
Deals with mathematics and a deeper understanding to provide more insight. Allows us to predict
Various business models:
Enterprise model
Data model
Behaviour model
Key communication elements:
Audience
Content
Attributes
Design and constraints:
Design is bounded by these constraints and the following consequence. Effective communication
clear.
"
(Reference: video of introduction to requirments)

"Interview Example Video: (ref: video of "interview example") (00:03:03)
Interviews are one type of risks. So they are given as example
Why are they considered as risks?
Interviews can occur at anytime. There can be no way to find out when they can come.
The result is we end up failing in interviews as we are not prepared. They are not like exams.

How to over come the risk of interviews?
There are two ways:
Risk mitigation
Contingency
"
(Reference: video of interview example)

"
MITIGATION :( ref: video of "Mitigation_Contigency") (00:03:00)
This can be understood when we talk about a business problem. We think that we are dealing with
software fails, it may lead to damages. So we need to deal with these unforeseen problems.
How do we deal with them??
We use mitigation. Mitigation means reducing the probability of damage happened by risks.
Example:
The fire alarms
Risk Statement : (ref: video of "Risk Statements")(00:03:24)
Risk statement is what we know define a risk in a project. Video talks about the importance of risk
when it occurs it damages the system or software. Training is not a risk because it can be predicte
because of the inaccurate assessments, poor project plans of building the software. We should ha
experimenting might help us.
Traffic Example: (ref: video of "Traffic Example") (00:02:58)
This example can give the better understanding of the analysis of how to proceed in building the so

This example can give the better understanding of the analysis of how to proceed in building the software
software work.
What can be the things we need to analysis?
Assess how early the person Is leaving for office.
Research on where the maximum time is taken because of traffic
Information on where the traffic jam can be there
How traffic changes with weather of the day.
Time taken for him with and with out traffic
"

(Referen ce: video of mitigation, risk statement, traffic exmaple)

--Last Saved--Dec 18, 2014 17:53:51
--Last Saved--Dec 19, 2014 16:20:47
--Last Saved--Dec 20, 2014 18:23:15
--Last Saved--Dec 20, 2014 18:27:40
--Last Saved--Dec 20, 2014 18:36:22
--Last Saved--Dec 20, 2014 18:39:46

Deliverables

| Delivearables | | | | |
|---|---|---|---|---|
| watch Intro to SE | understand the basics of SE | Understood the basics | 15 min | 35 min |
| Engineering notebook | watch engineering notebook | understood how to use engineering notebook | 22 min | 40 min |
| Test driven development | understanding what is TDD | Learnt different types of models in TDD | 22 min | 45min |
| state diagram | what and how to use state diargam | understood what are stagediagrams | 5 min | 15 min |
| requirments | introduction to requirments | learnt what requirments means | 33 min | 1hr 30 min |
| Interview example | use example to understand what is risk | understood what risk means and different ways to solve risk | 3 min | 10 min |
| Mitigation | understand what mitigation means | learnt what mitigation means using fire alarms | 3 min | 10min |
| risk statement | what is risk statement for a project | learnt to write risk statements | 4 min | 12 min |
| understand problem using example | learn how to solve traffic example | used traffic example problem to solve the probelm | 3 min | 8 min |

Lessons Learned

| Reflections | |
|---|---|
| took more time to understand every video that is watched | time taken to learn basics |
| completed according to schedule | understood risk assesment,  and basics of software engineering |
| ENB | |

Plan for next week

| Plans | |
|---|---|
| HINT | Complete hint functionality in quiz |
| Lives | implement Lives functionality |
| Creating Objects | implement objects in the core notifications code |
| redo btn | |