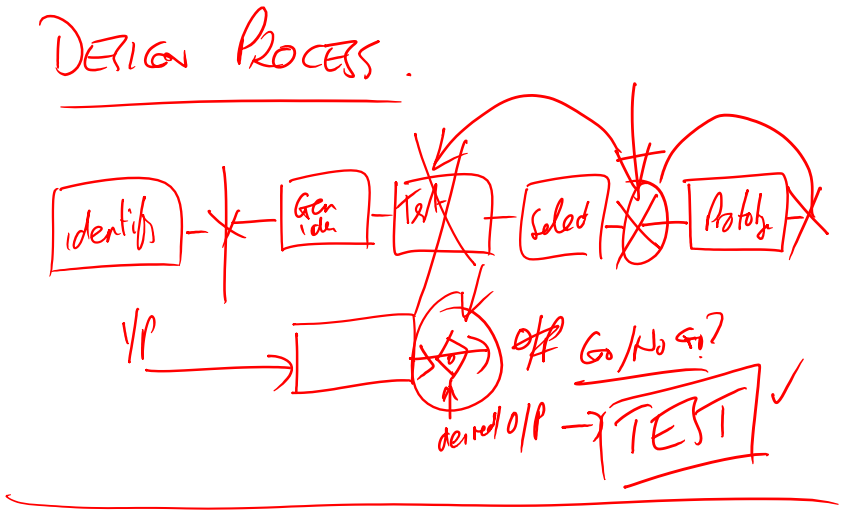
ECSE-211

Lecture 1415

8 February 2016

Design VI



Testing

- Tests of a prototype serve to confirm the validity of the entire design
- An effective, carefully considered test plan should include the following:
  - Statement of specific purposes of the tests.
    - What decisions will the test validate?
  - Specific test objectives
    - What needs to be measured during the test?

Testing

  - A step-by-step procedure for conducting the test
    - Attention should be paid to the variables to be controlled or monitored
  - An outline of the expected results
    - A data sheet with the predicted data outcomes
    - If you don't know roughly what should happen – how do you know if it is wrong?

Testing - Details

- So…
- A Test document should be created which describes each test
- Tests should be designed to examine potential weak points in the system design – given the design being considered.
  - E.g.
    - Does the ultrasonic sensor provide the readings predicted?
    - Can the robot find the goal, with an accuracy of positioning that is expected in the design for it to work?

Testing - Details

- Note – these are TESTS, i.e.; we know what the system must achieve to satisfy the specifications. Now we need to find out if it does.
- An EXPERIMENT would try to answer the question
  - How accurately does the ultrasonic sensor return values?
  - Under what conditions does it fail?
  - i.e. we are trying to characterize the system represented as a black box
- Tests should be designed to check extreme cases of the specifications
  - E.g. what happens if the ambient light is extremely high?
  - What happens if the robot is traveling fast and hits a wall or an object?

Testing - Details

- Tests should be designed to check extreme cases of the specifications
  - E.g. what happens if the ambient light is extremely high?
  - What happens if the robot is traveling fast and hits a wall or an object?
  - What is the worst case situation that can be encountered during the operation of the robot?

A Test Document

- Goal: Determine if the robot can avoid obstacles
- Result: It does
- Actions: None

A Test Document

- Goal: Determine if the robot can avoid obstacles
- Result: It does
- Actions: None

Can I repeat this?

- Is this a good test document?

A Test Document

- Goal: Determine if the robot can avoid obstacles
- Result: It does
- Actions: None

- Is this a good test document?
- What is missing?

A Test Document

- *Date:* 11 February 2015
- *Tester:* John Smith
- *Author:* John Smith
- *Hardware version*: robot version 2.1
- *Software version:* 3.2
- *Goal:* Determine if the robot can avoid obstacles

A Test Document

- *Procedure:* The robot should be placed at the origin of the grid facing North. Obstacles will be placed at grid locations (0,1), (1,0), (2,2). The robot will be instructed to travel to location (4,4). The test should be performed at least 10 times.
- *Expected Result:* The robot should identify any obstacles in its path and avoid them by at least 5cm. The robot LCD display should indicate that it has seen an obstacle. It should arrive at the destination with an accuracy of better than 5 cm.

Test Document

- *Test Report:* The test was performed 12 times following the protocol described above. The complete results can be seen in the spreadsheet (Obstacle_Avoid_Test_11Feb.xls). In summary, the robot hit an obstacle on 5 of the 12 runs. In each case, the robot approached the obstacle on a diagonal and hit a corner.

Test Document

- *Conclusion:* The robot performance did not meet the specified outcomes. The obstacle avoidance is unreliable.
- *Action:* This test report should be sent to the software team to review the obstacle avoidance process. The Gantt chart should be updated to show the revised tasks
- *Distribution:* Software development, project management

Sections in the Test Document

- Date, Tester and Author
- Versions of hardware and software being tested
- Purpose of Test
  - Explain briefly the overall purpose
    - Include – design version number, limitations on the scope of the test
- Test Objectives
  - Detail exactly what is desired from the test

Sections in the Test Document

- Test Procedure
  - Step by step instructions for carrying out the test
- Expected Results
  - Summarize expected outcomes from the tests
- Format of Output Required
  - Handwritten sheets, computer spread sheets, graphs, etc.
- Details of test carried out and results

Sections in the Test Document

- Conclusions: Pass/Fail?
- Actions to be taken as a result of the test
- Distribution list for the report

Testing

- Each subsytstem has a design specification
- Each subsystem has requirements documents
- Each subsystem has a design
- Each subsystem has a test set associated with it..
  - Each test has a document associated with it including the rationale for the test, the test protocol and the results

The Test Plan

- Just as with the design timeline, there should be a test timeline
- This should be integrated with the main design timeline
- Some of the design phases might be dependent on the successful completion of certain tests.
- *Develop a test plan!*
- Does this involve full regression testing?

Testing

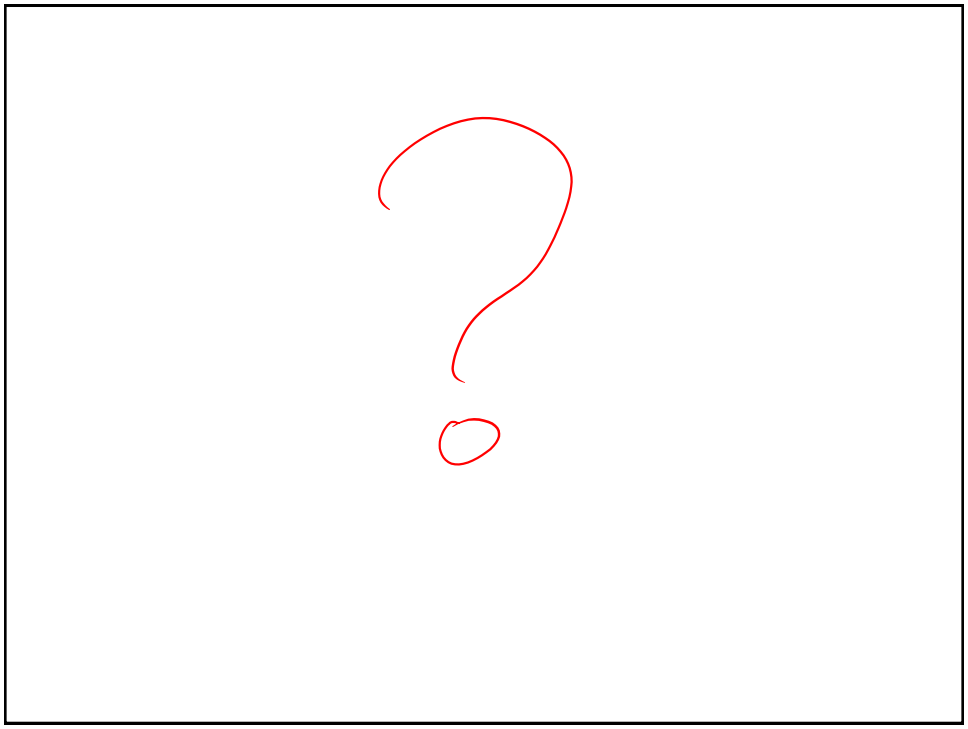
- Note – *Tests can consume an inordinate amount of time..*
- Incomplete testing can result in potential failure points not being identified.
- Badly designed tests achieve nothing and may result in failures further downstream in the process.

What is a failure?

- How do you define a test failure?
  - Is there an error bound on the compliance with the specifications?
  - What is it?
  - Is a failure a terminal event or can the system "soft fail", i.e. gradually degrade in performance?..
- How will your process respond to a failure?
  - Have you built this into the budget and into the timeline?

Testing

- This is as important as the design process
- It provides the information for controlling the design process
- Well designed test sets can catch problems early and reduce the costs.
- Testing gets more expensive as you move further into the design process
- ALLOW 2 WEEKS FOR THE FINAL INTEGRATION TESTS OF THE SYSTEM



Project Management

- We have a design process but to implement it needs management
- How do we control the process?
- There are several steps – points at which the results can be checked
  - Each check point is a go/no go decision
  - Some components can execute in parallel
  - The process must complete at a specific time

GANTT Chart

- Chart to illustrate a project schedule
- Work breakdown of project in terms of terminal elements and summary elements
- Can show current schedule status

- Dates back to around 1910
- Now implemented on personal computers and are common in collaborative projects..