# Week 11: Instructor Notes

## Overview

This week is a continuation of understanding end-to-end Testing by exploring the concept of operations and workflow testing. Different architectures pose different testing risks.

## Objectives

By the end of the week, each student will be able to:

* Differentiate testing techniques for different architectures
* Devise an End to End test plan

## Before the Week Begins (Prepare)

Preparation:

* Post Annoucement.
* Coordinate with TA.
  + Discuss struggling students and make plans to support and mentor.
  + Discuss W11 grading expectations.

Partnership/Group work:

* Students will be working on the Logging and Architecture approach to Testing.
* Check Canvas “groups” to verify that all students are enrolled in a group.

## During the Week (Teach)

Labs:

* Architecture Testing

## Looking Ahead

To be aware of

Next week's announcements:

* Please create and post your Announcement for W12 by Friday of this week.

# Prepare

## Overview

This week is a continuation of understanding end-to-end Testing by exploring the concept of operations and workflow testing. Different architectures pose different testing risks.

## Objectives

By the end of the week, each student will be able to:

* Differentiate testing techniques for different architectures.
* Devise an End-to-End test plan.

## Preparation Material

To be prepared for this module's activities, please read the following and be prepared to start or complete activities for Teach One Another and Prove assignments.

### Reading

* [See Reading.](https://webmailbyui-my.sharepoint.com/personal/wac3_byui_edu/Documents/GitHub/cse270-course/Reading/Reading.html)

As you read, try to answer the following questions:

# How do I test different architectures?

## Overview

Not all systems have the same type of software architects. Learning and identifying the testing needs of each is beneficial in making sure you are end-to-end Testing.

## Topics

Based on your reading, work in groups of 2-4 individuals.

Answer each of the questions from the list below. Answer the questions by researching the topic and provide a summary for each group of questions. Make sure you cite your sources; you need to have at least two others other than the readings.

Architectures

* Pick two of the software architectural terms.
  + Complete the following:
    - Defining each of the terms?
    - Which non-functional requirement would best test the architecture? ( See week 5: Reading Miguel’s Review of Quality Models)
    - How would you go about creating a test for it?
  + Terms:
    - Client-Server:
    - Component
    - Data
    - Event-driven
    - Layered
    - Peer-2-Peer
    - Rule-based
    - Service-oriented (SOA)
    - Web Services
    - Mobile
    - N-tier
    - Database
    - Event-driven
    - Cloud (PaaS, SaaS, etc):
* Research one of the types of flows identify, and answer the questions below
  + Flows
    - Data Flow
    - Process Flow
    - Workflow
    - Control Flow
    - Functional Flow
  + Questions
    - What is the purpose of the flow?
    - What insights does the flow provide to testing at the unit, integration, system or acceptance level?

## Submission

Create a word document that contains your

* Your group name and members
* Two architecture research, with associated answer summary
* Reach on Flows, and associated answer summary

**Rubric**

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Questions 70%** | Answered all question, clearly described and well supported with quotes | Answered all the question in detail. | Missing 1-2 answers, and answers are detailed. | Missing majority of answers to questions or answers consist of 1-2 sentences. | No answers to questions |
| **Team Work 20%** | Assignments have meeting time or team discussion summary | Assignments are identical | The team assigned different question to individuals and compiled into one document | The team assigned different questions to individuals and sent them in individually | The team worked on assignments individually and submitted them individually |
| **Professionalism 10%** | The paper is easy to read and communicated. | Properly cited, there are no grammar or spelling errors, and the writing style is "professional." | Found an instance of a spelling error, grammar error, incomplete citation, overly verbose wording, poor formatting, or poor writing. | A citation is missing where one is needed (plagiarism alert!). | Gross spelling/grammar errors or other aspects of the writing that make the paper difficult to read. |

The distribution of points starts at 50 percent for the minimal participation expectation and additionally stated expectations increase the percents.

# Prove: Logging

## Overview

With the advent of Object-Oriented Programming, Encapsulation and Data Hiding are key to reducing software code and complexity. On the flip side, it also presents a problem for Testing. The information is not as accessible nor available during development to find the results needed to prove a requirement, design, or concept. The development of well-managed logging is essential to Testing, debugging, operations, and maintenance.

## Backstory

To receive testing feedback on [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html)'s operations, both in the front-end and back-end software, log files are essential to track actions completed within the system.   
Using the communication paths found in the Lab, follow the instructions

* Communication between one [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) Device and another Device(Peer-To-Peer).
* Communication between one [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) Device and another Device(Peer-To-Peer).
* Communication between Transportation and [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) (Wireless).
* Communication between Home Base and [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) (Client - Server).
* Communication between Supply Chain and [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) (Rule-Based).
* Communication between User and [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) HMI(Event-Driven).
* Communication between [Dr. Mobile](https://byui-cse.github.io/cse270-course/Labs/DrMobil.html) to Database (Data).

## Instruction

To identify when things go wrong, you need to where and what is happening to the data, state changes, and connectivity. Good debugging, logs, alert messaging, and status updates can make a big difference.

### Pick five communication methods from the Background.

For each background communication path, use the 4 Use Cases to verify the results by identifying the type of logging information you would like to see between Dr. Mobile. :

* Use Case: Disconnection
* Use Case: Slow connection
* Use Case: Lost Data
* Use Case: Recovery

### Log Message

You will need to identify the different logging levels(ERROR, WARNING, DEBUG, INFO) in the log file, user, and if the system needs to do some other action. Also, identify the structure of the message.  
Pick one of the log types, and write a log message that can verify the use case.

* DEBUG: The finest grain logging level. Usually, a programmer using this logging statement to output primitive values, entrance and exiting of functions, and state change.
* INFO/STATUS: The INFO statements provide information about the general health of the system.
* ALERT: The ALERT statements inform the user that there is an action that is required.
* WARNING: The system is outside the normal happy path process and is trying to recover.
* ERROR: This is when the system has varied from the happy path, and the system can still recover.
* CRITICAL: This is when the system has tried a recovery but was unable at that location, and a reset of the states or program had to be completed.

### Format message

A good log message will contain a timestamp, specific information about the problem. Use your imagination to determine the values that need to be displayed and what you are trying to capture.

## Make it your Own

The completion of the core of this assignment is 89%. The assignment needs additional personalized work to achieve 100%.

Ideas for making it your own.

* Use another communication method.
* Come up with another use case.
* Determine another log state.

Create three to four additional log messages with your new idea.

## Submission

Make sure that you upload a copy of your document to iLearn. Upload your test code.

## Rubric

The following is the generic rubric for this assignment:

* 10 pts 1st Bug Report from your presentation using your form

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Logging Information 80%** | Make it your own | Identify the 20 different messages prototypes | Contains all 20 use cases (5 communication channels x 4 use cases) logging levels | Has two or more major issues | No answers to questions |
| **Professionalism 10%** | Make it your own | Properly cited, there are no grammar or spelling errors, and the writing style is "professional." | Found an instance of a spelling error, grammar error, incomplete citation, overly verbose wording, poor formatting, or poor writing. | A citation is missing where one is needed (plagiarism alert!). | Gross spelling/grammar errors or other aspects of the writing that make the paper difficult to read. |
| **Citations 10%** | One of the citations is a primary source | Contains 3-4 citations other than the reading. | Contains 1-2 more citations other than the reading | Contains citations from the reading | No Citations |