University of
South Australia

INFS 2044

Workshop 2b

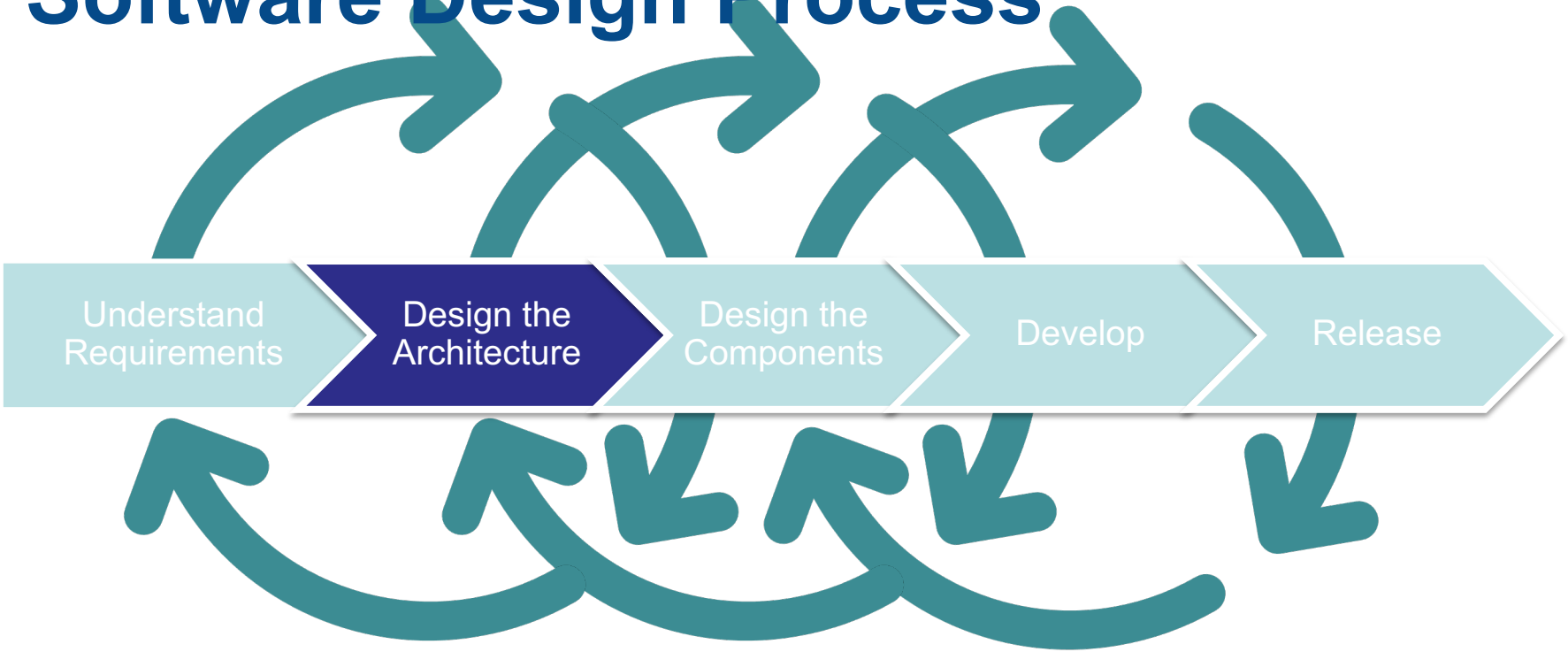
Preparation

- Read the required readings for this week
- Bring a copy of the workshop instructions (this document) to the workshop

Where We Are At

- Validated requirements and use cases (Week 1)
- Introduction to volatility-based decomposition
- Compositional design to realise use cases

Software Design Process



Learning Objectives

- Apply volatility-based architecture design to complex requirements

Task 1. Identify Volatilities

- Consider the *Sample Testing System (STS)* case study presented on the subsequent slides.
- Discuss potential volatility related to this system.
- What changes in the system and its environment may affect the design?

STS Requirements (1)

- The system should enable University *in-house Engineers* to:
  - Record the quality of product testing outcomes across the testing process
  - Analyse root causes of quality defects
  - Store data in a local database

STS Requirements (2)

- Test Engineers obtain product specimen from the production plant. There shall be a link between the specimen's test results and the data collected during its production.
- Test Engineers subject the specimen to testing using a mixture of manual and automated test stations. Some tests run for days and require multiple measurements before/after/throughout the test.
- The sequence of tests matters as some tests are destructive.
- Test Engineers record the test results obtained from several different test stations.
- Work process, data format, and access method differ for each test station.

Record Test Result Use Case

1. *Test Engineer* selects specimen and begins a new test.
   System records the commencement of the test.

2. *Test Engineer enters data.*
   System records the data and associates the data with the specimen.

Categories of Volatility

- User
- Client application
- Security
- Notification
- Storage
- Connection & Synchronisation
- Duration and device
- Workflow
- Locale
- Regulations
- …

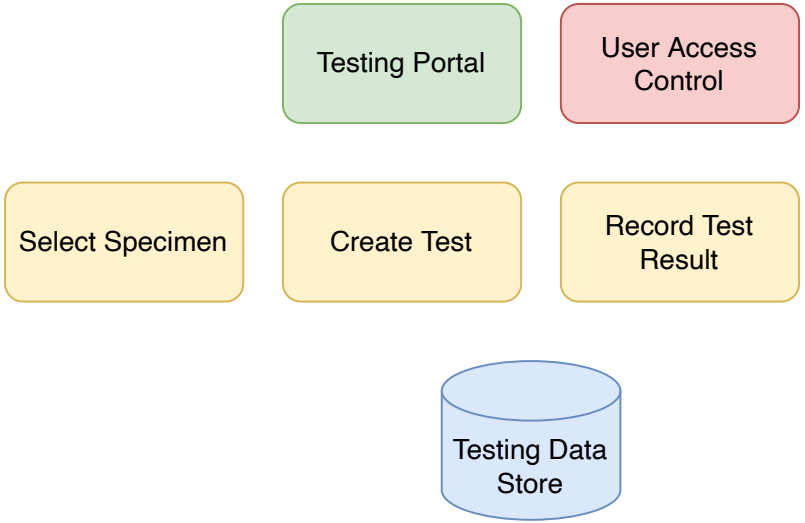
Task 2. Assess STS Decomposition

- Examine the decomposition given on the next slide.
- Discuss advantages and disadvantages of this design.
- How would changed requirements affect the design?

Recall STS Requirements

- Record the quality of product testing outcomes using a variety testing activities
- Connect to and ingest data from a variety of test stations
- Link test outcomes to the specimen’s production run

STS: Design 1


Testing Portal
User Access Control
Select Specimen
Create Test
Record Test Result
Testing Data Store

Task 3: Component Design

- Create a decomposition for the STS that accounts for the identified volatilities.
- Show how the volatilities map to components.
- Identify strengths and weaknesses of the decomposition.
- Does it isolate change and promote evolution and reuse?

Task 4: Validation

- Validate the architecture by creating a Communication Diagram or a Sequence Diagram for use case *Record Test Result*

You Should Know

- Identify volatilities in system requirements
- Identify components based on volatility and design principles
- Validate a component design on use cases

Activities this Week

- Complete Quiz 2

University of
South Australia