Department of Electrical, Computer, and Software Engineering

Faculty of Engineering and Applied Science

SOFE 3650-Fall 2021

Software Design and Architectures

Project Description and Deliverables

This document presents a description of the project expectation and deliverables.

Prepared by: Dr. Ramiro Liscano
Date: October 2021

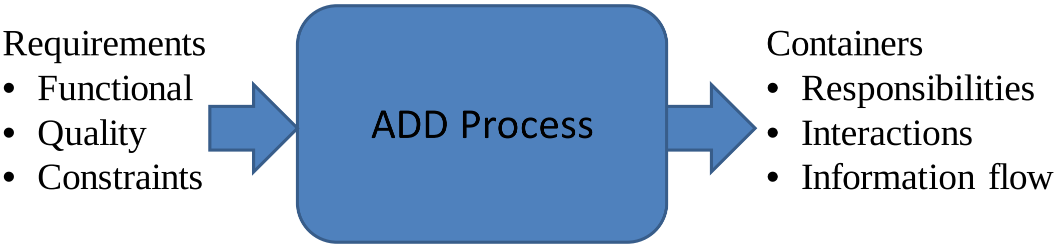
SOFE 3650 Project Deliverables Oct 2021
Author: Dr. Ramiro Liscano

Objectives

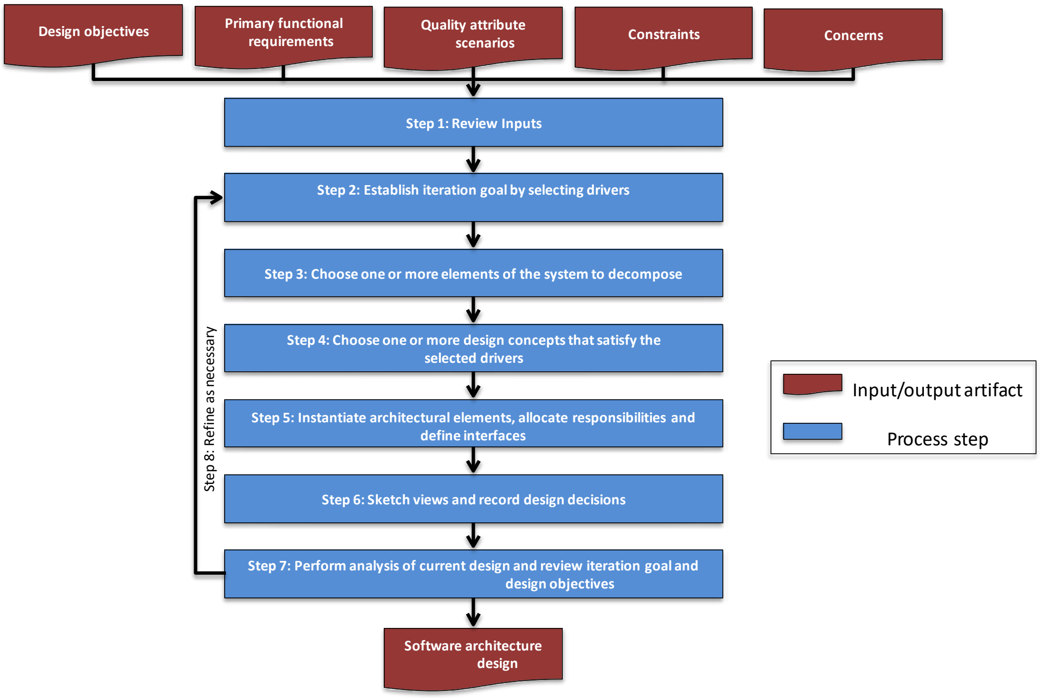
The objective of this project is to demonstrate a methodological set of steps in the design of a software architecture for a set of requirements provided by your instructor. The expected design approach to take is the Attribute Driven Design (ADD) presented in some detail in the text book [1] and briefly overviewed here.

The ADD process

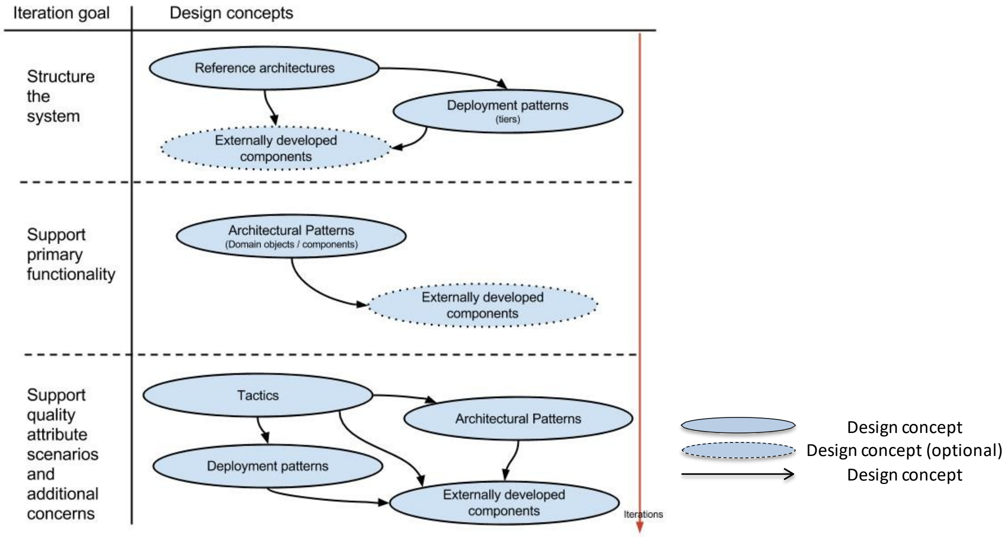
The ADD process takes a set of requirements as inputs and expects an architecture design and description as output as shown in the diagram below.



The ADD is a 7 step process as shown in the figure below with the expectation of 2-3 iteration steps of the process.


Design objectives
Primary functional requirements
Quality attribute scenarios
Constraints
Concerns
Step 1: Review Inputs
Step 2: Establish iteration goal by selecting drivers
Step 3: Choose one or more elements of the system to decompose
Step 4: Choose one or more design concepts that satisfy the selected drivers
Step 5: Instantiate architectural elements, allocate responsibilities and define interfaces
Step 6: Sketch views and record design decisions
Step 7: Perform analysis of current design and review iteration goal and design objectives
Step 8: Refine as necessary
Software architecture design
Input/output artifact
Process step

One of the most common types of software systems is the greenfield systems in mature domains type of system, that is a new system but based on existing architecture patterns and styles. In this type of system the norm is to perform 3 iterations of the ADD method with the respective goals of i) defining the structure of the system, ii) defining the components of the architecture to achieve the functional requirements of the systems, and iii) refactoring the architecture to achieve the quality requirements of the system. These goals and design concepts for each iteration is shown in the figure below.


Iteration goal
Design concepts
Structure the system
Reference architectures
Deployment patterns (tiers)
Externally developed components
Support primary functionality
Architectural Patterns (Domain objects / components)
Externally developed components
Support quality attribute scenarios and additional concerns
Tactics
Architectural Patterns
Deployment patterns
Externally developed components
Iterations
Design concept
Design concept (optional)
Design concept

Deliverables

The expectation is to submit a set of artifacts in a GitHub classroom repository that your instructor will create for you that demonstrates the ADD steps as applied to the design of an architecture for a student defined project.

All deliverables should be uploaded to a GitHub repository that the instructor will set up. Individual contributions in the project will be graded the based-on GitHub commitments of the deliverables and use of the KanBan project board available in GitHub. A good way to manage your project is by leveraging the Issues tagging available through GitHub and the automated KanBan project board.

Deliverable 1 – Project proposal (Due Oct 18)

Project group members should submit a project proposal preferably of some software project they are familiar with such as one they have already worked on in other courses. This deliverable should contain title of the project, list of group members, and a set of functional and non-requirements.

Deliverable 2 – Project progress report (Due Nov. 4)

Prior to commencing the ADD process the use cases, quality attributes and constraint requirements of the project need to be defined based on the requirements that you submitted. The format of these submissions should follow closely the FCAPS case example in the text book [1].

Expected artifacts are:

- Use Case model
- Quality Attributes for the use case
- System Constraints for the use case

Deliverable 3 – Design of the Use Case (Due Dec. 6)

This portion of the project should follow the ADD design process. The format of the submission should mimic the FCAPS case example in the text book [1]. All 3 iterations of the design process are expected and all 7 steps should be included.

Grading Scheme

Proposal: 2 marks
Progress Report: 3 marks
Design Submission: 25 marks
Presentation: 5 marks.

Team Assessment

As a requirement of the course there is a team assessment that should be completed. The purpose of this team evaluation is for students to understand the dynamics and roles of a team. This will be assessed using the ITP metrics tool and instructions will be posted in Canvas.

References

[1] “Designing Software Architectures: A Practical Approach” by Humberto Cervantes and Rick Kazman and covered in the course.