Software Engineering

WEEK 02 LECTURE 02 A

TODAY

- Vision of the product.
- Scope of the product.
- Scope Creep Situations.
- Strategies to Handle Changing Requirements.
- Requirement and Design

VISION AND SCOPE
OF SOFTWARE PRODUCT

VISION

- When a product is designed, it is guided by a principle known as product vision.
- The **product vision** is what outlines the value of a product to the client and its place within the wider market.

VISION

- It relates to the purpose of the product, as well as the problem or need the product sets out to solve.
- Any changes that occur in the project should fall within the vision of the product. In other words, changes to the project should not change the purpose of the product.

SCOPE

- Visions can become problematic if they are not realistic. **Scope** is defined as what a project can realistically achieve.
- It is an important aspect of the project that has the power to greatly affect development teams and product managers.

SCOPE

- Defining scope is an important part of the project, and it is best done during the requirements elicitation activity of the project.
- It is also important to manage client expectations during this process and not to over-promise what the development team can realistically deliver.

SCOPE

- A good strategy for managing expectations is to identify what will not be included in a product, so client expectations are clear.

SCOPE AND VISION

American software engineer, consultant, and author Karl Wiegers has defined both vision and scope. The definitions are:

> **Vision** is a "long-term strategic concept of the ultimate purpose and form of a new system." (Wiegers, 2012, p. 1)

> **Scope** is what "draws the boundary between what's in and what's out for the project." (Wiegers, 2012, p. 1)

We can see that the two concepts are interrelated and why scope is so important to project development. Defining the scope ensures that the project undertaken is not only possible but also successful.

SCOPE CREEP

- **Scope creep** happens when the scope of a project grows as the result of an increase or change in requirements. Consequently, the likelihood of project success usually significantly lowers over time.

AVOIDING SCOPE CREEP SITUATIONS

HOW TO HANDLE CHANGES
AND

- Changes are common to software requirements and need to be accounted for.
- Strategies for defending against scope creep include:

- Making expectations clear between client and team.
- Drawing the scope with the client through tools such as use case diagrams.

- Asking the client to prioritize requirements so that the most important ones can be developed early.

- Asking “Is this in scope?” when refining requirements to avoid spending excess time working on unnecessary requirements or capabilities better suited to later releases.

- Providing concrete time frames for developing requirements with clear beginning and end dates also helps determine whether the requirement is realistically doable and within the scope of the project.

- Estimating time frames is a skill that software product managers refine and improve with time.