

IFB295 Study Guide | 2022 Semester 2

Mark Walpole | Notes for IAB330 at the Queensland University of Technology

Table of Contents

- [IFB295: IT Project Management](#)
 - [Week 1](#): Project Management, User Stories, Scrum, and Teamwork
 - [Week 2](#):
 - [Week 3](#):
 - [Week 4](#):
 - [Week 5](#):
 - [Week 6](#):
 - [Week 7](#):
 - [Week 8](#):
 - [Week 9](#):
 - [Week 10](#):
 - [Week 11](#):
 - [Week 12](#):
 - [Week 13](#):
-

IFB295: IT Project Management

In your information technology career, you will participate in and then lead project teams that are expected to deliver real benefits to stakeholders. This unit builds on your previous studies of earlier units to define a high-level solution by using a range of approaches of project management methodologies and frameworks. You will enhance your learning of these approaches (Agile, PRINCE2 etc.) by practicing it collaboratively with other students. To be successful in a complex environment, you need to employ appropriate project management strategies, tools and techniques for a given context. This unit provides you with the necessary knowledge and skills to enable you to effectively

manage your project in the IFB398 Capstone Project (Phase 1) and IFB399 Capstone Project (Phase 2) and to be prepared for a project environment in industry.

Week 1: Project Management, User Stories, Scrum, and Teamwork

Project Management

Project management is the act of planning, organising and creating procedures and protocols to ensure that a project achieves its goals, is delivered on time and is of a high quality.

Terminology

- Methodology
 - A methodology is a set of methods, principles and rules that follow concepts such as paradigm, theoretical model, phases and techniques.
- Framework
 - A framework is a basic structure underlying a system, concept, or text
- Artifact
 - A by-product produced during the development. This could be a use case diagram, class diagram or other design documents that help describe the design and architecture

Scrum

Scrum is

Scrum Roles and Responsibilities

- Product Owner
 - This is the owner of the product. They are responsible for optimising value, ordering the product backlog in priority and product incrementing
- Scrum Team
 - The scrum team is the team working on the project. They are responsible for self-organisation, cross-functional, ordering and choosing which items appear in the sprint backlog
- Scrum Master

- The scrum master is the leader of the scrum team. They are responsible for coaching, training, and supporting the scrum team.

User Stories

User stories are short informal descriptions explaining a software feature from the perspective of an end user. It's purpose is to demonstrate how the software or a feature will provide value to the customer. User stories also help define the acceptance criteria of a particular feature.

For example:

As a [role], I want to [do / see / change something] so that [outcome].

INVEST

All user stories should follow the INVEST acronym.

- Independent
 - User stories should be independent as dependencies can make planning, prioritisation and estimation difficult
- Negotiable
 - User story details should be worked out in a conversation between the customer and the developer
- Valuable
 - The user story must provide some sort of value to the customer. User stories should be written with the customer to ensure this
- Estimable
 - The feature or product the user story is crafted around should be estimable. These estimates are then used in the initial prioritisation and planning of the project
- Small
 - User stories should be short and sweet. They must only represent a few days in person effort as this allows them to be more precise and accurate
- Testable
 - User stories should be able to be crafted into acceptance criteria. We can't develop things we can't test

Acceptance Criteria

Acceptance criteria is the criteria that must be met before we can say a feature is completed. They should follow a Given-When-Then template such that:

- Given some context
- When some action is carried out
- Then a particular set of observable consequences

For example:

As a lecturer, I want to be able to see a list of all students enrolled in my classes so that I can see class lists and numbers enrolled.

Acceptance Criteria:

- **Given:** I have a "Show Classes" link displayed for the classes I lecture
- **When:** I click on "Show Classes"
- **Then:** The system should display the list of classes together with class activity, class numbers, day and time, and room where the class is held

Epics

Epics are large user stories that are too big to implement in a reasonable timeframe. These types of user stories are split into smaller separate stories to help spread and breakdown the workload.