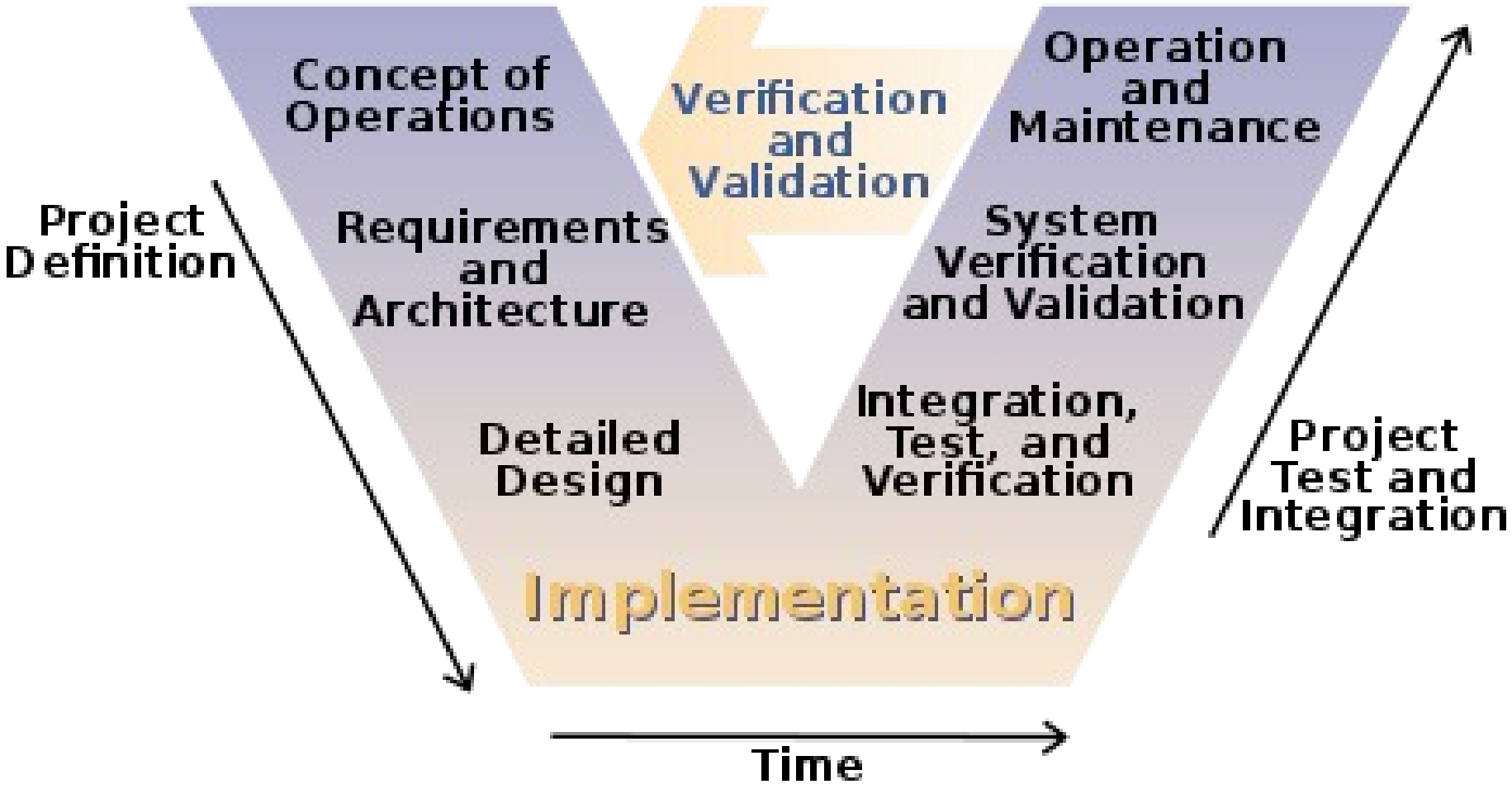
V-Model Presentation

Presentation for the class of ESOF - 3MIEIC05

T2 - Software Processes

Duarte Nuno Esteves André Lima de Carvalho– up201503661@fe.up.pt
João Álvaro Cardoso Soares Ferreira – up201605592@fe.up.pt
João Augusto dos Santos Lima – up201605314@fe.up.pt

What Is The V-Model?



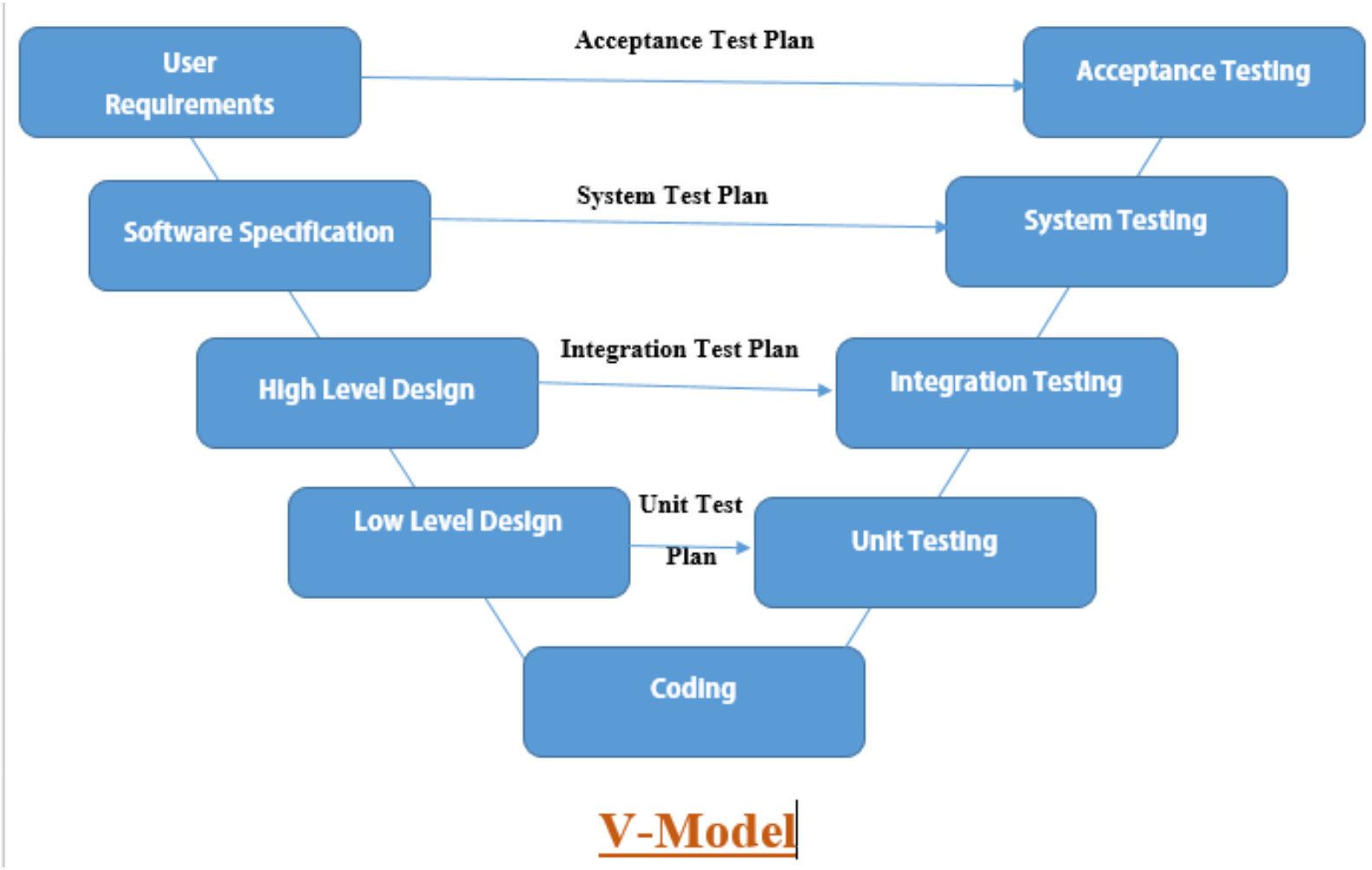
What Is The V-Model?

- **V-Model is a strict model for software development developed as an extension from the Waterfall model.**
- **It can be summarized by its shape, a V:**
  - The left side has the verification stages.
  - The right side is comprised of the validation stages.
  - At the point where both sides meet, at the bottom of the V, code is developed and implemented.
  - The tests on the right side are developed at the same time as the respective concepts on the left side.

Origin of the V-Model

- The V-Model, or Vee Model, was created by the German government in the 1990s to manage project development in their defence department.
- It was developed as a counter-part to the PRINCE2 model, and is to this day still used by the UK and USA's governments.

The Stages



V-Model

The Stages – Left Side (Verification Stages)

- User Requirements – Where the user's needs and respective solutions are determined.
- System Design – Where the techniques and methods that will be needed to accomplish the solutions are decided.
- Architecture Design – Where the high-level concepts for the code are designed, making use of tools such as class diagrams.
- Module Design – Where the functions to be developed in the implementation stage are defined.

Implementation

The middle-point of the V and usually the most time consuming stage, this is where the code is developed.

The Stages – Right Side (Validation Stages)

- Unit Testing – the goal of this phase is to get rid of bugs at a code level, being the lengthiest of the testing stages.
- Integration Testing – checks if all components are working together properly.
- System Testing – often developed while in contact with the consumer, these tests verify if the entirety of the system is working as expected and as was designed.
- Acceptance Testing – the last overall stage, it checks if all the needs of the client are met.

Pros and Cons

Pros

- Simple to use and understand, especially for smaller projects.
- Prevents extensive debugging, as all issues are discovered at an appropriate stage.
- There’s an overwhelming amount of testing, documentation and stability due to the highly strict approach, leading to a more stable product.

Cons

- Lacks flexibility and ways to respond to unexpected problems during development.
- Promotes strictness and streamlining, sometimes overlooking creative solutions.
- Issues with time constraints.
- Ill-suited for continued support after launch (for example, with patches).
- Due to how easy it is to understand at first, inexperienced developers might stick to its guidelines too much.

When to use the V-Model

The V-Model is ideal for projects where stability is a priority, at the cost of a higher budget and lengthier development time, along with a product that won't need frequent patches.

Examples of V-Model Usage

The V-Model is frequently used in software for:

- Healthcare
- Military
- Government

Sources

A Software Process Development, Assessment and Improvement Framework,for the Medical D evice Industry - Dundalk Institute of Technology

What is V-Model? - airbrake.io

V-Model vs Scrum: Who Wins? - reqtest.com

Good Design Practice for Medical Device and Equipment - Cambridge University

Using V-Model for Testing - Carnegie Mellon University