

**NAME**

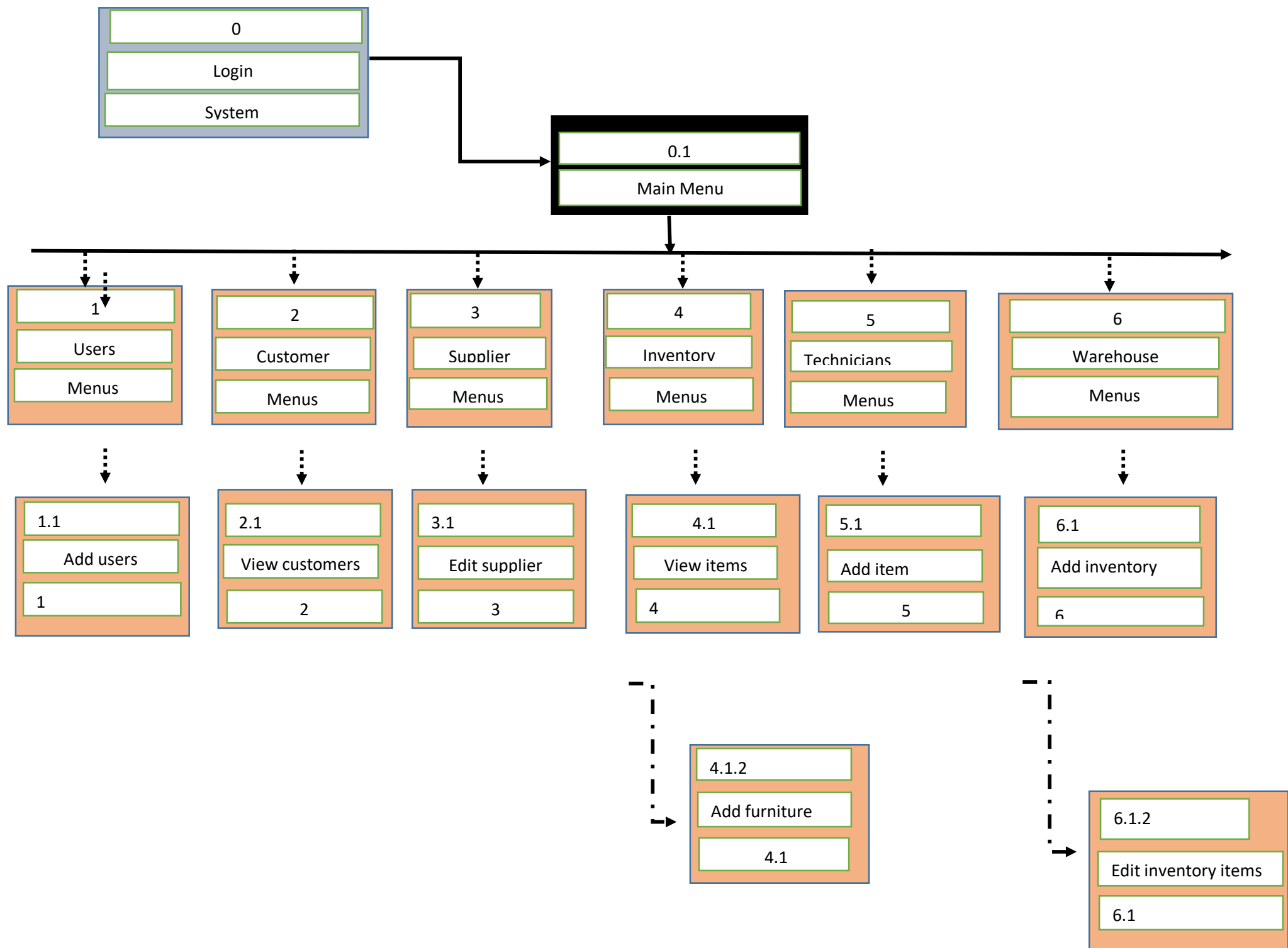
**COLLEGE NUMBER**

## Introduction

Dialogue flow diagrams (DFD) are essentially used in system design and development to help systems stakeholders have a rough or a quick overview of the system to be implemented. Usually this is blue print for the engineers and the systems owners. End users can also use the DFD to grab a blind interaction of the expected deliverables within the final application whenever need be. It's also vital to detect bugs, loopholes and potential risks within the DFD. During development, and depending on the specific architecture and software methodology chosen, the DFD is implemented phase on phase, process on process till the final product is achieved. Further, the DFD shall borrow all the components that are described from the ER diagram initially. These components are also called entities, upon which the database structure is built.

In this case, we consider the case of Pine valley furniture that has abstracted into the following entities;

- Furniture
- Supplier
- Inventory
- Customer
- Technician
- Payments
- Users
- Warehouse



## Software Installation

There are three methods to install the software, though each depends on the platform that is supported the needs, cost allocations and requirements and the ability of skilled labour variable. Installation can occur in the following areas:

- Mobile apps
- Web apps
- Desktop application
- Console application

The deployment process and approaches

In the case of a web app like this system being designed and developed for PVF, there are three options to consider:

1. A cloud server
2. A local host
3. Converting the web app into a desktop app and installing on the desktop

## Cloud servers

Cloud servers are remote server applications that hold and store information remotely. They can be accessed by shell applications or file management applications like Putty. Examples include Amazon, Google cloud platform or Microsoft Azure.

Advantages

- Secondary servers are frequently supported and maintained to date
- They are more secure
- They have 24 hour support system

Limitations and risks

- They may be expensive in the long run
- Since programs are mutually shared, data may leak to on shared server

- Require specialised personnel to operate

### **Localhost**

A Web app is developed and managed on the local machine or repository, together with an apache/nginx server.

#### Advantages

- Easier to manage
- Quick and effective for local administration
- Is cheaper to maintain
- Does not require any special skills

#### Limitations and risks

- It's not secure as anyone with access to the machine can access the software
- Relies on one machine to be uptime most of the time

The approach on the kind of software installation will depend on the following:

**Cost:** Does the organisation have enough funds to facilitate the hosting process

**Skills and technical:** Checks if there are enough man power with relevant skills to support the app

**Infrastructure:** Checks to see if the organisation has enough resources to build, deploy and maintain the app