Design document

**Name: Tony Jiang**

**Date: 19 Sep 23**

**Project: Video call system**

|  |  |  |
| --- | --- | --- |
| **Versions** | **Date** | **Description** |
| **0.1** | **19 Sep 23** | **Initial document** |
|  |  |  |

Contents

[Introduction 3](#_Toc146208681)

[Context 3](#_Toc146208682)

[Goals 3](#_Toc146208683)

[Scope 3](#_Toc146208684)

[Functionality 4](#_Toc146208685)

[User Interface Design 4](#_Toc146208686)

[System Architecture 4](#_Toc146208687)

[Level 1: System Context (C1) 4](#_Toc146208688)

[Level 2: Containers (C2) 5](#_Toc146208689)

[Level 3: Component (C3) 5](#_Toc146208690)

[Level 4: Code (C4) 5](#_Toc146208691)

# Introduction

This document covers all the technical aspects of this project, including its structure and the design decisions made throughout the development. It helps to further reflect on the design choices within this project. By doing so, it helps with planning how everything should be configured and address any potential development challenges. Additionally, it helps in conveying the intended design to other developers, ensuring a shared understanding and agreement on the design approach. Some information may refer to other documents where the information is fully detailed. New content or components will be added in this document throughout the project, and it is intended for technical developers.

The project's objective is to create two or three solutions for a video call system within the PRAS application.

# Context

This is to give general information about the project. What’s the project about and from whom. For context, please refer to the **Project Plan** document, on ‘*1.1 Context*’.

# Goals

This is to give information over what the project is trying to achieve and what the purpose of the project is. For the goals of this project, please refer to the **Project Plan** document, on ‘*1.2 Goal of the project*’.

# Scope

This is to define what needs to be delivered for the entire project, to ensure clarity for all stakeholders. The scope of the project can be found in the **Project Plan** document, on ‘*1.3 Scope and preconditions’*.

# Functionality

This is to define the core functionality of the system so that it’s clear to everyone. To achieve a clear understanding of the functionality of the system, we use something called ‘user story’. User story describes the functionality of the product, the expected behavior, and the required components. You can find the user stories in the **User Story** document.

# User Interface Design

The purpose of this is to present the user interface to all the stakeholders and reach an agreement on the product’s visual design. To illustrate this is we make use of a wireframe, which are mockup design of an application or website. Think of wireframes as the initial sketch when you're creating a painting; they lay out what will appear on the canvas before you start painting. There are websites or applications that allows you easily create wireframes for free. You can create, export, demonstrate, and get feedback on it instantly. With the wireframe you can also provide detailed descriptions for each component on the wireframe. The wireframes can be found in the **Wireframe document**. Tools for creating the wireframe: [Figma](https://www.figma.com/files/recents-and-sharing/recently-viewed?fuid=1217416791179059026).

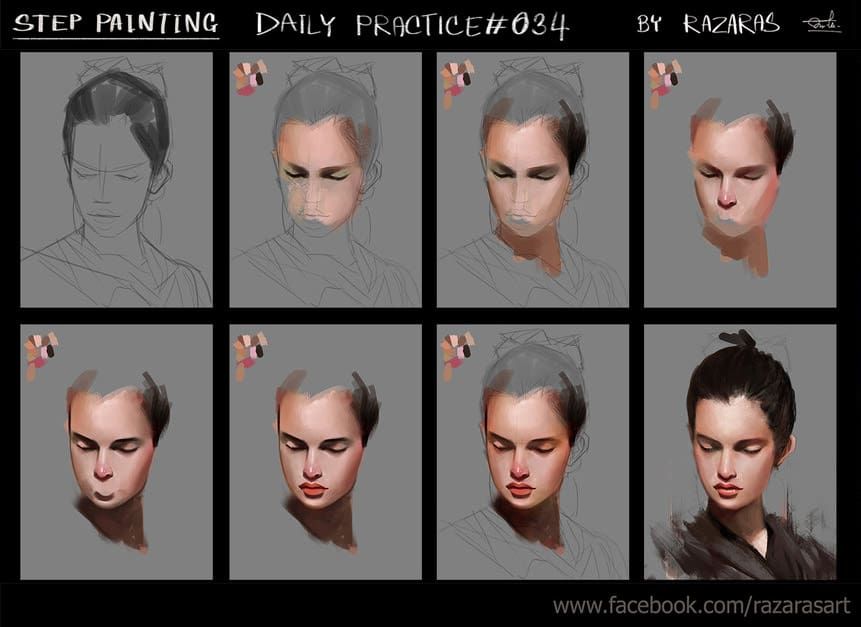


Figure : Sketch painting example

# System Architecture

The purpose of System Architecture is to describe the internal system’s overall structure and establish an agreement on the desired design of the system. To make it easy to describe and communicate the system’s architecture, we’ll use a C4 architecture diagram. It’s an architecture design that way easy to understand. The design approach is straightforward and helps us communicate how each part of the system should be setup, even to a non-technical person. It’s like using Google Maps and you’re trying to zoom in on a place in Aruba. The closer you zoom the more details you see. It’s like that. The C4 architecture diagram helps us do the same for our software’s structure.

The C4 architecture diagram has 4 levels:

* Level 1: System Context (C1)
* Level 2: Containers (C2)
* Level 3: Component (C3)
* Level 4: Code (C4)

We will go to each level and describe what they do when we reach there. The tool that was used to create the C4 model is [Visual Paradigm online](https://online.visual-paradigm.com/diagrams/features/c4-model-tool/). The free version.

## Level 1: System Context (C1)

At this level you’ll discover which intended user is going to use the system and what the system does. This provides an overview of the entire system. Additionally, it helps to describe what the system does to non-technical person.

A screenshot of a computer

Description automatically generated

Figure : C1

In the C1 model, you can see that employees can make video calls to the retirees using the video call system. This should give you an idea of who can use the system, what system it is and what do they do on the system.

## Level 2: Containers (C2)

This is the second level after we zoomed in to the system context level 1. This is to see the high-level technical building blocks (container) that makes up the system and what the relationship between them are. The containers represent an application or database use to build the entire system. From this level and onward, it’s not meant for non-technical person.

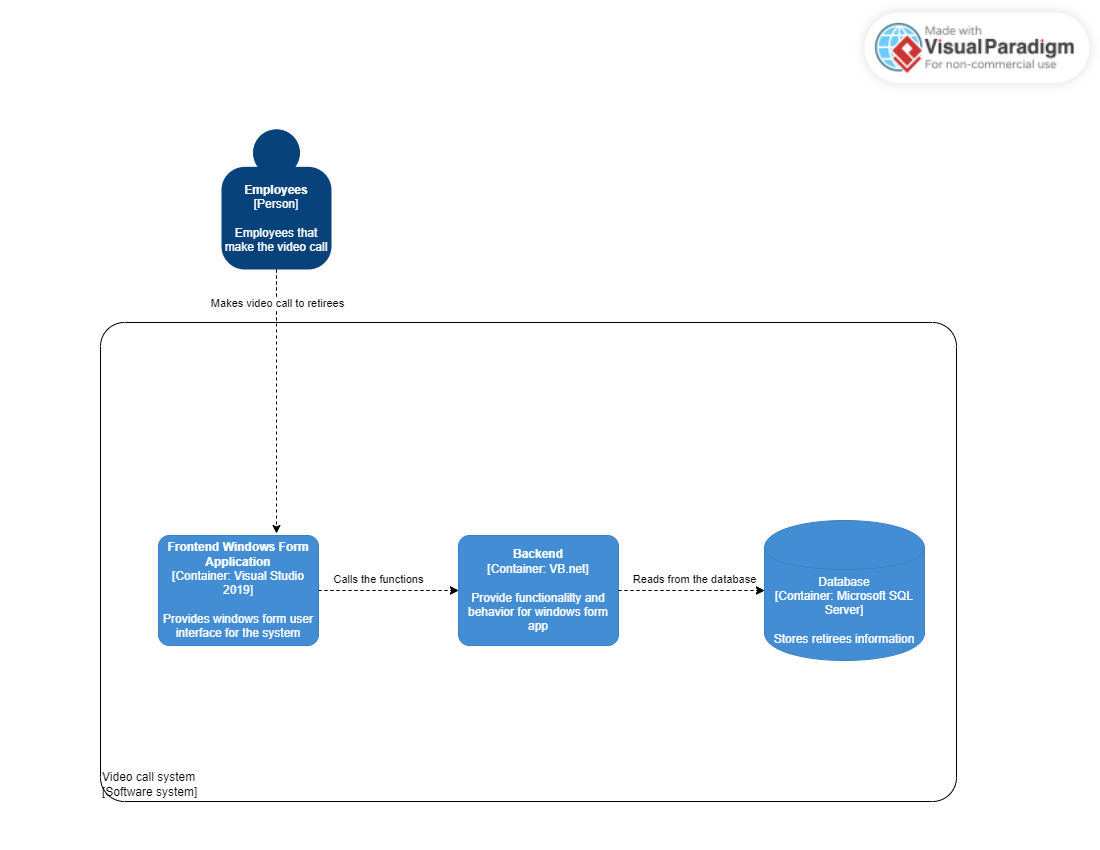


Figure : C2

It’s pretty understandable how the system is put together from the C2 model. There are 3 containers that makes up the system.

* **Frontend Windows Form Application**
* This is the frontend part of the system, where the user interface resides.
* It calls the backend part of the system when users interact with the frontend.
* The PRAS application uses Visual Studio Windows Form Application.
* **Backend**
* In here reside all the functional code to make the application do things.
* It gets and passes information from the frontend.
* It retrieves information from the database to the frontend.
* **Database**
* It stores all the information from the retirees.
* It passes the information needed to the backend to use.

## Level 3: Component (C3)

## Level 4: Code (C4)