RoSys Program Manual

# Introduction

This manual is a guide, that should help clarify some of the ideas we used throughout the program. It should detail how some parts of the program works.

An important thing to keep in mind about our program, is that some parts, such as the user interface, where made purely to demonstrate the programs functionality. The main part of the program, was the “Core” project, which is meant to serve as a backend server to any/multiple user exposed interfaces.

Our focus with the Core is intergrateability into other systems, so we made sure to use interfaces for some of the main resources with the program.

This focus on being able to integrate the program, as well as wishing to demonstrate the programs functions, on its own, was a greater challenge than expected.

# The User Interface

Remember: The User interface was made purely to demonstrate features of our program

## Initial Run

When running the program, you initially get 3 options. This is essentially our “login”, where you select which type of user you wish to interact with the program as.

The different types of users have some different functionality. Student and Teacher generally have the same features, but Teacher have access to more rooms, here-included meeting rooms.

Admin has some management features, like deleting reservations, registering new rooms, and access to book even more rooms, and selecting specific rooms for the reservation.

# Code

This section will cover some of the features of the code.

## Core: System Settings

The System Settings class, is currently used to define the “environment” the program runs in.

We set up 3 levels of environment, but only uses 2: Development and Production currently has no differences, but Test affects which database the connection runs to.

### Update System Environment

This method updates the Environment Variable in the DAL.

It is run when the Env property is set, and under Initialize.

## Core: HelperFunctions

Used to keep various generic functions that are used throughout the core project.

# UI.GUI: LoggedIn

Is the class that stores which user is currently logged in. It is used by the GUI to keep track of which user was initially selected.