Shady Dell Design Brief

Client Information

Client Name: Shady Dell Elementary School
Client Industry: Public Education

Project Overview

Shady Dell Elementary School is seeking to establish a directory management system for it's students, faculty, staff, and administrators. The primary goal of this project is to provide an efficient and flexible directory management system for the school.

Requirements

- As a User, I want to be able to create and store information about individuals in my organization.
    - Requirement: Persistent information storage
    - Requirement: Data models individuals
    - Requirement: User can upload new data
- As a User, I want to be able to quickly search through my personnel information
    - Requirement: User can query information about personnel
    - Requirement: User can see personnel data.
- As a Developer, I don't want to fight spaghetti code for three years
    - Requirement: Use some sort of modular software architecture.

Development Schedule

Using Agile Story Points (4-12 hours), the schedule is as follows:

- Design Database: 1SP
- Implement Database: 1SP
- Create DAL: 1SP
- Create Utilities: 0.5SP
- Create Models: 0.5SP
- Create Controller: 1SP
- Create View: 2SP
- Test Product: 1SP (Keep this number by using TDD in other stages)

A team of 3 developers may accomplish this in 3-9 working days

Zachery Smith – CIS 170

This is the estimation of a standard system with a fully functional GUI.

Deliverables

Interview Transcripts (Complete)

Requirements Documents (Complete)

Design Brief (Complete)

Demo (Complete)

Product

Tested Product

Deployed Product

Data and Logic Model

Menu

Handles user input and display

AddPeople()

Collect Information
Construct Person
Use PersonController to save
rinse, repeat

PersonModel
Represent a Person

Used as a "common language" for the data in all parts of the program

SearchPeople()

Collect Information
Query people using PersonController
Output Result

PersonController
Perform Data Manipulation
Interact With DAL

Store the people during program execution

FileAccess DAL
Store and retreive data from the JSON file

JSON FILE