Team Rain Delay

Emergency Services Directory
Software Architecture Document

Version 1.2

Revision History

Date	Version	Description	Author
	1.0	Initial Version	
19/May/08	1.1	Filled in All Missing Sections	TJ Wasik
20/May/08	1.2	Added reference to the template for this document Added comments on architecture section	TJ Wasik

Emergency Services Directory	Version: 1.1
Software Architecture Document	Date: 19/May/08

Table of Contents

Software Architecture Document

1. Introduction

1.1 Purpose

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

1.2 Scope

This document applies to the overall design of the system. It contains information relating to the architectural design of the software, the Structure of the Database, and of the physical servers hosting the site.

1.3 References

The template for this document:
http://web.cs.wpi.edu/~gpollice/cs4233-a05/rup_sad.html

1.4 Overview

This Document provides the reader with a deeper understanding of the design of the Emergency Services Directory Website. It is organized into several views of the system, the Logical view, which will give an overview of the subsystems and components the system is comprised of, the Deployment view, which will give go into how the system has been deployed and what it's being hosted on, the Implementation view, which will show how the system has been implemented and go into more detail about the different parts of the system, and the Data view which will show how the database is structured and organized.

2. Architectural Representation

This system was designed using a standard three tiered architecture, with a Presentation Tier, Application Tier and Data tier. The Presentation Tier contains all of the visible web pages and handles all input from and output to the user. The Application Tier handles all of the Business logic and provides an abstraction to the database, the Data tier consists of the Database and stored procedures contained within, this provides the persistence required for the system. The Logical View shows a quick overview of all of the basic subsystems in the System and gives a basic overview of the System as a whole. The Deployment view tells how the system is physically configured. The Implementation View gives a more in-depth view into how the system has been implemented. The Data view shows how the Database is setup and structured.

3. Architectural Goals and Constraints

The previous two teams had done a considerable amount of the design work. As a result, we were required to use the existing architecture. We didn't have much time for any re-factoring work and decided instead to focus on the addition of new features, such as more accessible and easier to use searching and Google Maps mashup.

4. Logical View

4.1 Overview
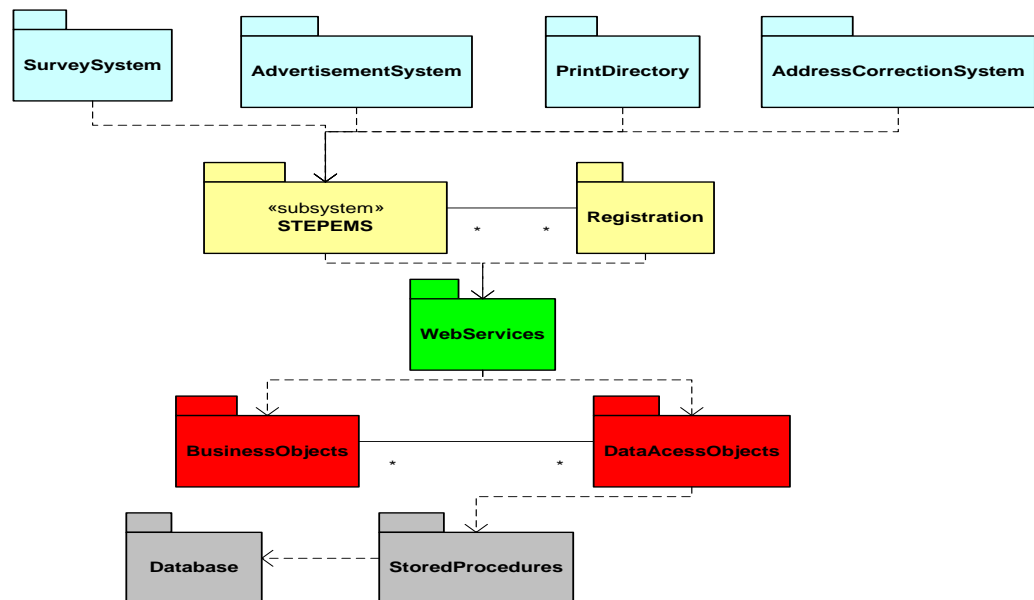
This Diagram shows the various high level packages that the system is broken down into. The packages can be seen here separated into the various layers by color, the Presentation layer is the blue and yellow packages, the Application layer is the green and red packages while the Data layer is the grey packages.

4.2 Architecturally Significant Design Packages

4.2.1 Application Layer

The Application Layer provides the business logic and connects the Presentation Layer to the Database. The Application Layer is contained within the EMSWebServices package. All communication with the Presentation layer is done through Web Services.

1.1.1.1 Web Services

This Package contains all of the Web Services that are provided by the application layer and consumed by the presentation layer

1.1.1.2 Business Objects

This package contains all of the objects which provide the Business logic for the system.

1.1.1.3 Data Access Objects(DAO)

This Package contains all of the objects which are used to communicate with the database. Each object contained in this package has a table or tables representing it in the database.

1.1.2 Data Layer

The Data Layer consists of the MS SQL server 2000 database and the stored proceudres contained within. The Data Layer provides persistence for the system and all communication is done through SQL queries, stored procedures and views.

4.2.2 Presentation Layer

The Presentation Layer is composed of two basic packages, Registration and STEPEMS. All front end components used Active Server Pages as the primary interface.

1.1.2.1 STEPEMS

This package contains almost all of the visible pages and the logic to do basic parsing and validation of data related to the pages directly. An example of some of the Subsystems contained within STEPEMS are:

1.1.2.1.1   Survey System

This subsystem is responsible for managing survey question and responses.

1.1.2.1.2   Searching

This subsystem is responsible for search queries.

1.1.2.1.3   Advertisement System

This subsystem is responsible for managing the advertisements in the system.

1.1.2.1.4   Print Directory

This subsystem is responsible for generating the Indesign file for the printed directory.

1.1.2.1.5   Address Correction System

This subsystem is responsible for the automatic correction of addresses.

1.1.2.2 Registration

This subsystem contains the components for the registration of new accounts and organizations.
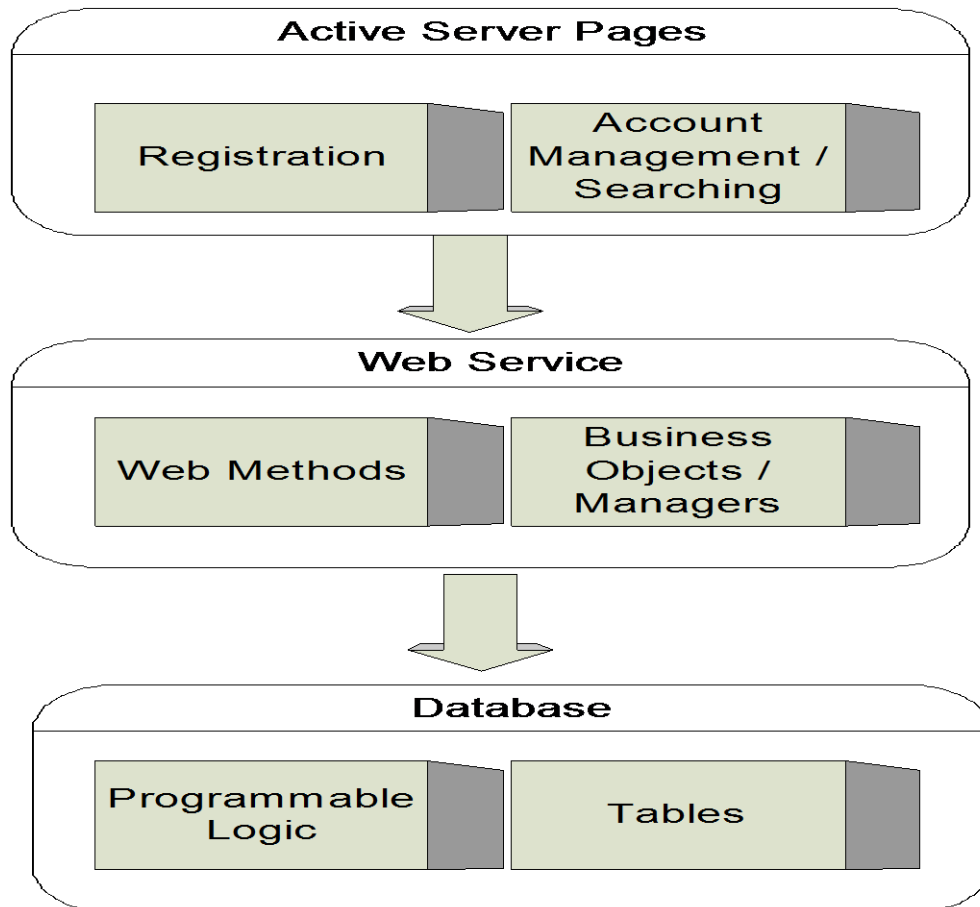
5.   Deployment View

This system is deployed on a single server hosting the database, webserver, and webservices. Clients connect using web browsers through the internet.

6. Implementation View

6.1 Overview

The system is implemented using a three tiered architecture. The first layer is the Presentation Layer consisting of Active Server Pages, This layer contains all of the pages that the user sees, and some of the basic logic to present them and validate input. The second layer is the Web Service layer, this layer provides all Business Logic functions and connects the Presentation Layer and the Data Layer. The third Layer is the Data Layer, this layer consists of the Database, and utilizes stored procedures to keep the logic of maintaining the database closer to the data.

6.2 Layers

1.1.3 Presentation Layer

The presentation Layer Consists of the STEPEMS and Registration projects. These projects contain all of the visible web pages, as well as some of the logic to deal with parsing and verifying user input. The Registration project deals only with the registration of a new user, or organization, and duplicates a lot of the functionality of the STEPEMS package. Both of these projects communicate with the Web Services layer, and gain access to persistent data through that layer.

1.1.4 Web Services Layer

The Web Services Layer consists of the EMSWebServices project and provides all of the Business Logic of the system. This layer also acts as an intermediary between the Data layer and the Presentation layer. This layer provides Web services which the Presentation layer consumes, and uses SQL queries to communicate with the data layer

1.1.5 Data Layer

The Data Layer consists of the MS SQL database and provides persistence for the system as well as the logic directly related to the manipulation of that data through the means of stored procedures. This layer communicates with the Web Services Layer by responding to SQL queries.

7. Data View (optional)
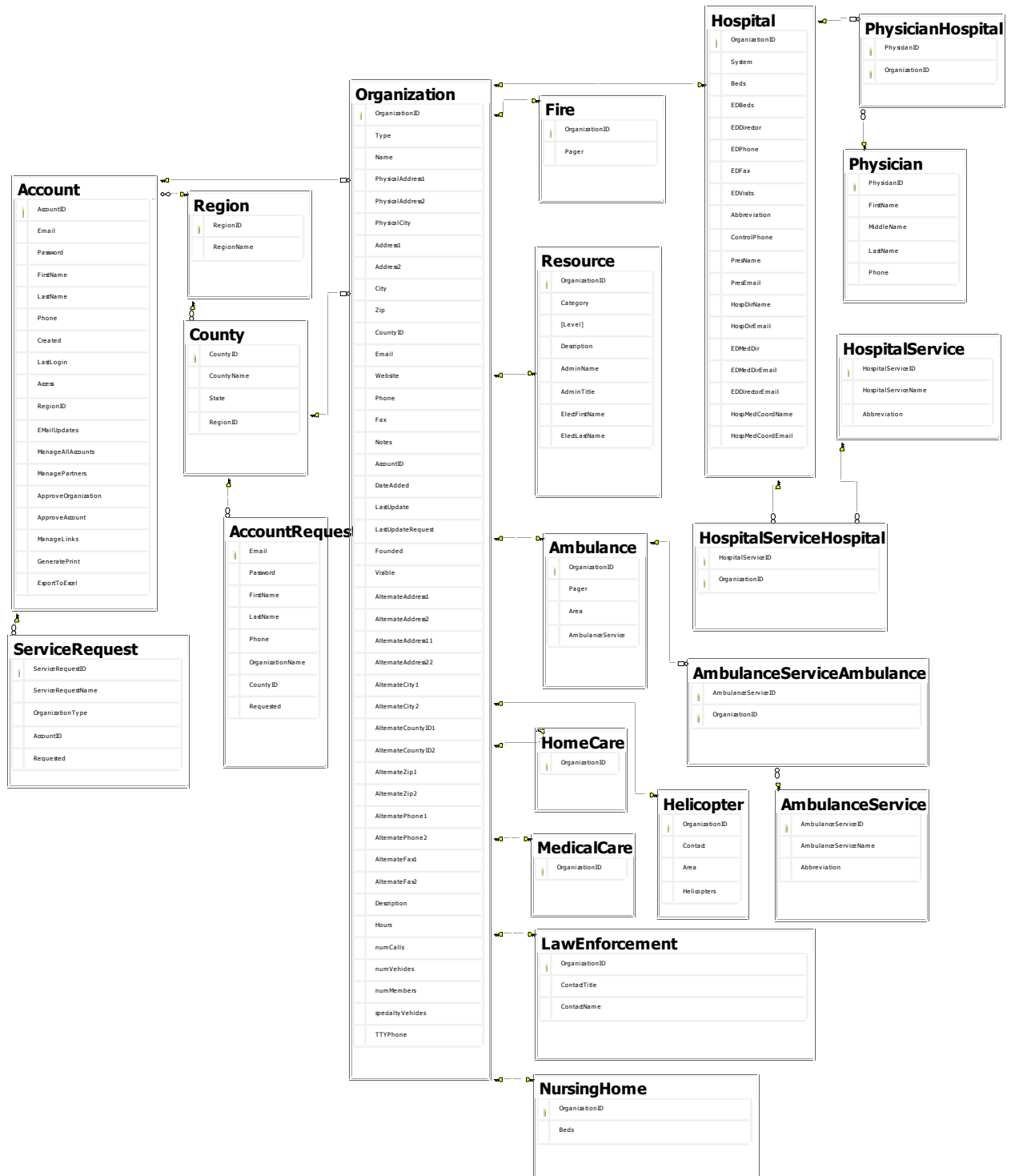
This Diagram shows the tables in the database of the system. As can be seen Organization is the primary table for an entry, each organization also has a sub-type which corresponds with each of the sub-type tables which are on the right side of the diagram. There are also tables which contain static data, such as counties. Users are stored in the Account table and are associated with a region.

8. Size and Performance

This system currently has mostly listings for the finger lakes region, but the editors are attempting to grow it first to the state level and then eventually to the national level, as such the system had to be designed to allow it to scale up to significantly larger numbers of users and registered organizations.

1. Comments on Architecture.

While working with this system we discovered many areas of the architecture that could use improvement, as such we are recommending that future teams consider a major refactor. The conceptual overview of the design, as a three tiered architecture is sound, though the implementation of that could use some work. The main area of focus for a refactor should be the presentation layer, consisting of the STEPEMS and Registration packages. These packages should be combined into one single project, as it is currently these packages have a lot of replicated functionality and create a large amount of complexity by being separate, while bringing in minimal benefits of this separation. These packages should also be inspected for updateability, as they currently are a simple, and likely to be performed update such as adding in a new type of organization is a difficult and time consuming task, which involves changing code in many places, a future team should consider finding a way to make this type of update simpler. There is also some application logic in this presentation tier that should be moved to the application tier, and some logic is actually split between the C# back-end and the ASP.net pages, the ASP pages should focus solely on presenting the information, all logic should be moved to the C# codebehind pages, with it split as it is now understanding how things work is difficult, and changing them is even worse.

The Application Layer, consisting of the EMSWebServices package is fairly solid. This package is well designed and well implemented, in a refactor this package would require little attention, it will probably require little re-implementation and just some rearranging of where some of the logic is located.

The Data Tier is solid and doesn't require any refactoring, the only changes that are likely to be required to the database are changes that are required to match up with the refactoring of the rest of the system.