Software Design

Specification

ADOC

(Automatic door open and close system)

**Table of Contents**

Table of Contents ii

Revision History iii

Approved By iii

**1. Introduction** 1

1.1 Purpose 1

1.2 System Overview 1

1.3 Design Map 1

1.4 Supporting Materials 1

1.5 Definitions and Acronyms 1

**2. Design Considerations**  1

2.1 Assumptions 1

2.2 Constraints 2

2.3 System Environment 2

2.4 Design Methodology 2

2.5 Risks and Volatile Areas 2

**3. Architecture**  2

3.1 Overview 2

3.2 Subsystem, Component, or Module 1 …N 2

3.3 Strategy 1…N 3

**4. Database Schema**  3

4.1 Tables, Fields and Relationships 3

4.1.1 Databases 3

4.1.2 New Tables 3

4.1.3 New Fields(s) 3

4.1.4 Fields Change(s) 3

4.1.5 All Other Changes 3

4.2 Data Migration 4

**5. High Level Design** 4

5.1 View / Model Element 1…N 4

6. Low Level Design 4

6.1 Module 1…N 4

7. User Interface Design 4

7.1 Application Controls 4

7.2 Screen 1… N 4

**Appendix A: Project Timeline**  5

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Name** | **Reason For Changes** | **Date** |
| *1.0* | *Sue Dillehay*  *David Kurth*  *Jon Fletcher* | *Initial Revision* | *8/19/2004* |
|  |  |  |  |

**Approved By**

*Approvals should be obtained for project manager, and all developers working on the project.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Signature** | **Department** | **Date** |
| Bill Currie |  | BP-IT-Development |  |
|  |  |  |  |

# **1.Introduction**

## **Purpose**

This design will detail the implementation of the requirements as defined in the Software Requirements Specification.

## **System Overview**

This project is required to develop GUI enabled real-time automatic **door open and close system** for CC3 building.

The doors can be in the following states: closed, currently opening, currently closing and open. The change of state, each time the controller button is pressed is given below:

doors closed - doors currently opening,

doors currently opening - doors currently closing,

doors current closing - door currently opening

and door open - doors currently closing.

## **Definitions and Acronyms**

Acronyms and Abbreviations:

a. “ADOC “ stands for automatic door open and close system.

b. SDS: Software Requirement Specification Definitions:

a. Concept drift :-concept drift means that the statistical properties of the target variable, which the model aims to predict, change over time unexpectedly.

b. Data Sampling:- Reducing the skewness of a particular class or set of classes in a dataset by means like upsampling,downsampling,oversampling etc.

1.4 ABBREVIATIONS

**1.5 Summarise**

“SDS” is aimed to display the transition from each state based on the input value of the person being approaching, crossing, or leaving.

# **2. Design Considerations**

This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution.

## **2.1 Assumptions and dependencies**

**ASSUMPTIONS:**

1)model is fully trained

2)data stream flow is continuous

3)outlier limit is low

4)performance matrix is suitable

5)100% accuracy of the system

**DEPENDENCIES**:

1)applicable data stream should be there

2)proper availability of net connection

3)Possible and/or probable changes in functionality

## **2.2 Constraints**

1)Non-Availability of continuous data streams can be major limitation to the system

2)Network communications

**2.3 GOALS AND GUIDELINES**

With the increasing popularity of the Internet-of-Medical-Things (IoMT) and smart devices, huge volumes of real time or IoT data have been generated.

So,the goal of **ADOC** is to display the transition from each state based on the input value of the person being approaching, crossing, or leaving fluently.

**2.4** **Design Methodology**

This model will be IOT based. The voluminous data stream will flow through the model continuously during the real time. ADOC will be active throughout and check the transitions.

If the performance of the model drops below the performance threshold, then the concept drift condition has taken place and so there is a need of re-training and relabelling the model.

## **2.5 Risks and Volatile Areas**

1)frequent change in transitions may cause bad results

2)an outlier’s mistake may risk the model

# **3. Architecture**

The architecture provides the top level design view of a system and provides a basis for more detailed design work

Provide or reference a detailed description and diagrams of the architecture..

## **3.1 Overview**

This section provides a high level overview of the structural and functional decomposition of the system. Focus on how and why the system was decomposed in a particular way rather than on details of the particular components. Include information on the major responsibilities and roles the system (or portions) must play.

## **3.2 Subsystem, Component, or Module 1 …N**

You only need to provide this level of detail for elements which are custom for this design. Do not go into gory detail. Goal is to get 80% of the elements figured out ahead of time.

Describe an element (subsystem, component, module, etc.) from architecture in further detail. When appropriate, include information on how the element is further broken down and the interactions and relationships between these subcomponents.

## **4. Strategy 1…N**

*Describe the strategy used or decision made. Include information on the alternatives considered and the reasons for their rejection.*

# **5. Database Schema**

## **5.1 Tables, Fields and Relationships**

*Table transition\_record and Numeric\_record are used to detect the change in drift.*

### **5.1.1 Databases**

*Mimic-2 is our given dataset.it is divided into transition and numeric database*

### **5.1.2New Tables**

*transition\_record:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |

*Numeric\_record:*

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

### **5.1.3 All Other Changes**

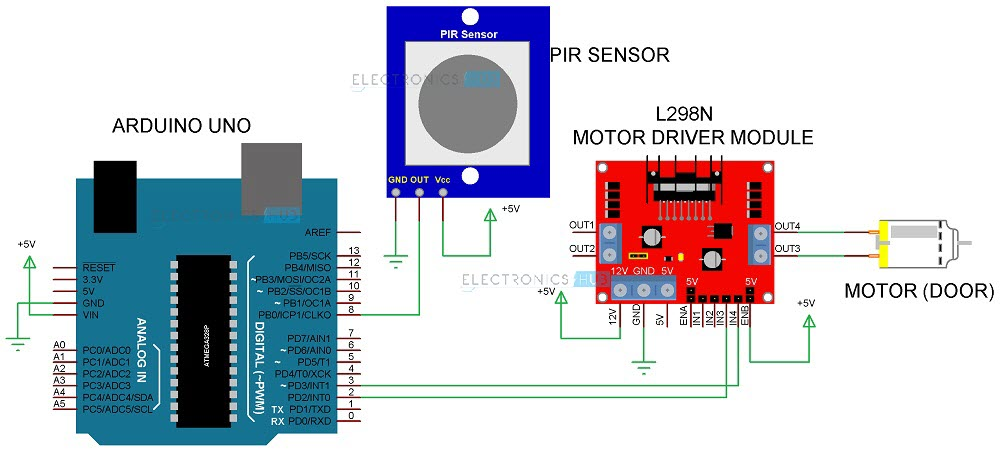
*If any other changes are requested (stored procedures, indexes, relationships, security settings, DTS packages, maintenance plans, etc), please describe what is needed here.*

## **5.2 Data Migration**

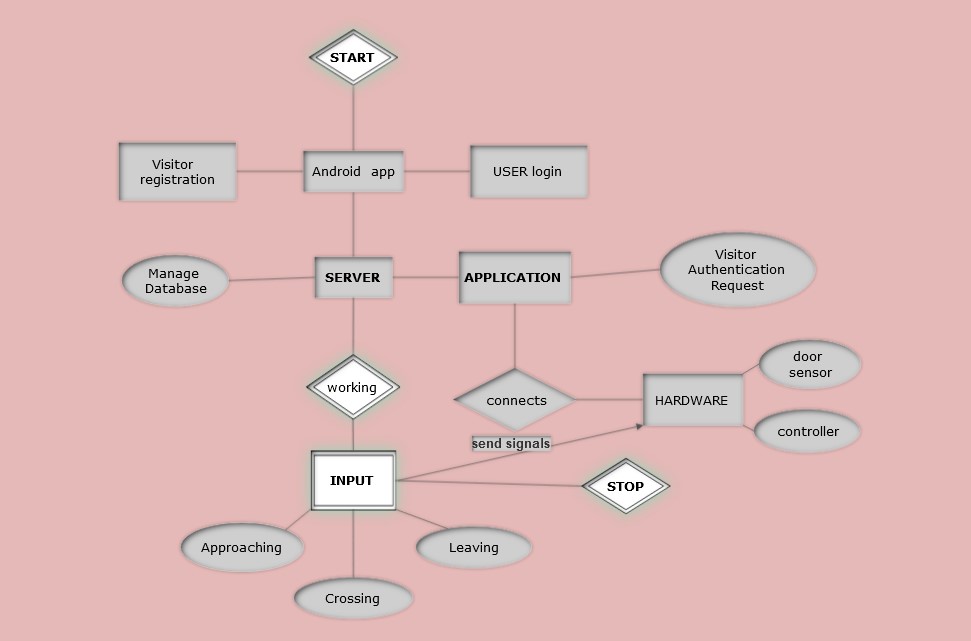
*(Optional) - Provide a description of how existing data should be migrated to new tables and fields.*

# **5. High Level Design**

5.1 Model overview



5.2 Working overview



5.3 Workflow sub-processes

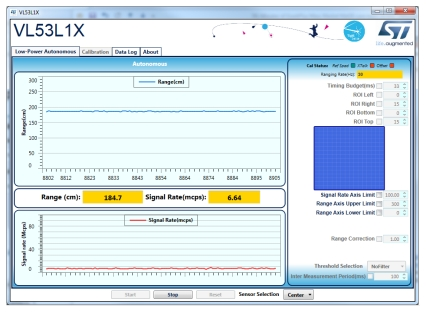
Workflow subprocesses will be created for each individual binder in each binder request. TheTemplate Model that will be used to create the subprocess is called BinderPerCarrier.Binder Extension sub processes will be created each time an extension is needed for each binder.The Template Model name used to create the subprocess is called BinderExtension.

**6. Low Level Design**

This section provides low-level design descriptions that directly support construction of modules. Normally this section would be split into separate documents for different areas of the design.

**7. User Interface Design**

This section provides user interface design descriptions that directly support construction of user interface Screens.



7.1 Application Controls

Detail the common behavior that all screens will have. Common look and feel details such as menus, popup menus, toolbars, status bar, title bars, drag and drop mouse behavior should be described here.

7.2 Screen 1… N

Illustrate all major user interface screens and describe the behavior and state changes that the user will experience.

Appendix A: Project Timeline

Reference the Microsoft project Binder Request Release 2 – Development..