**Requirements Analysis Document (RAD)**

**Data Analysis Subsystem (DAS)**

**for the**

**Predictive Wastewater Management System (PWMS)**

**Phase 1.0**

Prepared by

Fairleigh Dickinson Software Development - Analysis Team

**Security Notice**: The information contained within this document is Restricted. Unauthorized disclosure is prohibited. Failure to observe policy regarding proprietary information can result in disciplinary action, including dismissal, as well as result in a violation of proprietary rights and subject you and/or third parties to legal liability.

Purpose

This document is intended as a “package” agreement between development groups

Document Control

Document History

|  |  |  |
| --- | --- | --- |
| Date | Version | Comments |
| 2/28/15 | 1.0II | Initial Version |
|  | 1.0 | Document placed under formal internal control |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Authors

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Title | Organization | Phone |
| William Phillips | DAS lead | FDU | 201-247-2573 |
| William Phillips | FDU faculty | CS Department | 201-247-2573 |
|  |  |  |  |
|  |  |  |  |

Reviewers

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Title | Organization | Phone |
| Mark LaPlante | Product Manager | Eastech | 918-664-1212 |
| William Phillips | FDU Faculty | CS Department | 201-247-2573 |
| Amitha Peddi | SQA Engineer | FDU SQA | 336-848-8346 |

Sign Off List

|  |  |  |
| --- | --- | --- |
| Name | Title | Phone |
| Mark LaPlante | Product Manager | 918-664-1212 |
| Alfredo Tan | CS Department Chair | 2324 |
| Amitha Peddi | SQA Engineer | 336-848-8346 |

Document Sign Off

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Table of Contents

1 Introduction 6

1.1 Purpose of the System 6

1.2 Scope of the System 7

1.2.1 Actors 7

1.2.2 Subsystems 7

1.2.3 Context Diagram 8

1.3 Objectives and Success Criteria of the Project 9

1.4 Definitions, Acronyms and Abbreviations 9

1.5 References 9

1.6 Overview 9

2 Current System 10

3 Proposed System 11

3.1 Overview 11

3.2 Functional Requirements 11

3.3 Nonfunctional Requirements 11

3.3.1 Security 11

3.4 Scenarios 12

3.4.1 Arun Creates an Algorithm 12

3.4.2 Sirisha Executes an Algorithm 12

3.5 Use Cases 13

3.5.1 Administer DAS 13

3.5.2 Authenticate and Authorize Data Analyst 14

3.5.3 Data Analyst Creates Algorithm 15

3.5.4 Data Analyst Executes Algorithm 16

3.5.5 Use Case Diagram 18

3.6 Object Model 19

3.6.1 SAInterface (Boundary) 19

3.6.2 DataAnalystInterface (Boundary) 19

3.6.3 ACSDASInterface (Boundary) 19

3.6.4 DASDSSInterface (Boundary) 19

3.6.5 AASDASInterface (Boundary) 19

3.6.6 FrameworkBoundaryComponents (Boundary) 19

3.6.7 SystemAdminControl (Control) 19

3.6.8 DataAnalysisControl (Control) 19

3.6.9 PredictiveAlgorithm (Control) 19

3.6.10 ConfigurationData (Entity) 19

3.6.11 AlgorithmStorage (Entity) 20

3.6.12 AlgorithmFactory (Entity) 20

3.6.13 FrameworkEntityComponents (Entity) 20

3.6.14 Conceptual Class Diagram 21

3.7 User Interface – Screen Mock-Ups 22

4 Dynamic Model 23

4.1 Administer DAS 23

4.1.1 Sequence Diagram 23

4.1.2 Communication Diagram 24

4.2 Authenticate and Authorize Data Analyst 25

4.2.1 Sequence Diagram 25

4.2.2 Communication Diagram 25

4.3 Data Analyst Creates Algorithm 26

4.3.1 Sequence Diagram 26

4.3.2 Communication Diagram 27

4.4 Data Analyst Executes Algorithm 28

4.4.1 Sequence Diagram 28

4.4.2 Communication Diagram 29

# Introduction

This document describes the requirements, the analysis functional model for the Data Analysis Subsystem (DAS) of the Predictive Wastewater Management System.

## Purpose of the System

The purpose of the Data Analysis Subsystem is to provide an integrated framework that an algorithm developer (Data Analyst) can use to create predictive (and other) algorithms intended to use, as input, data provided by the Data Storage Subsystem. Interface objects shall be provided in order to efficiently access Data Storage subsystem data as well as providing an interface to send outputs to an output interface/boundary object as well as to the Audit and Alert subsystem. An interface to the Access Control Subsystem (ACS) shall be provided in order for the algorithm designer to secure the execution of the algorithm.

In analysis modeling terms, the Data Analyst will create a control object (i.e. an algorithm) that, when run, will receive inputs from the Data Gathering/Data Analysis boundary object and will provide outputs. In other words, the Data Analysis subsystem is comprised of Boundary and Entity Objects. Control objects shall be created/emitted from the subsystem that can be run standalone. Authentication and authorization may optionally be provided by the Data Analyst when the algorithm is designed. An interface to the Access Control subsystem shall be provided to support this capability.

## Scope of the System

### Actors

|  |  |
| --- | --- |
| **Actor Catalog** | |
| **Actor Name** | **Description** |
| Data Analyst | An actor that performs data analysis by creating and using adaptive optimization algorithms on the data managed by the PWMS |
| System Administrator | The System Administrator is an actor who adds and deletes users to and from the PWMS, assigns roles, and performs other system administration duties |

### Subsystems

|  |  |
| --- | --- |
| **Actor Catalog** | |
| **Subsystem Name** | **Description** |
| AccessControl(ACS) | The PWMS subsystem that assigns roles and maintains access control lists (ACLs) for the system actors and enables users to assume roles for the PWMS. Authorize and authenticate actors to access the PWMS subsystem. |
| AuditAndAlert(AAS) | The PWMS subsystem that periodically surveys the PWMS data and alerts the Watchman of a potential emergency situation. Maintains a store of audit records for the PWMS for an audit report. |
| DataStorage(DSS) | The PWMS subsystem that stores and organizes the PWMS data for use by the DataGathering, DataAnalysis, and AuditAndAlert subsystems. |

### Context Diagram



## Objectives and Success Criteria of the Project

A review checklist shall be created for this RAD document. Success of this project or shall be contingent upon the checklist criteria being met to the satisfaction of Eastech and Fairleigh Dickinson

## Definitions, Acronyms and Abbreviations

DAS – Data Analysis System

PWMS – Predictive Wastewater Management System

RAD – Requirements Analysis Document

TBD – To be Determined

## References

System Level Requirements Analysis Document for the Predictive Wastewater Management System (PWMS)

Bernd Bruegge and Allen H. Dutoit, *Object-Oriented Software Engineering: Using UML, Patterns and Java*, 3rd

Edition, Prentice Hall, 2010 (ISBN 0-13-606125-7)

Jacobson, Ivar *Object-Oriented Software Engineering-A Use Case Driven Approach.* Addison Wesley 1992.

Martin Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language,* 3rd ed., Addison

Wesley, 2003.

Roger S. Pressman, *Software Engineering: A Practitioner’s Approach*, 7th Edition, McGraw Hill, 2009 (ISBN

9780071267823)

Black and Veatch, Smart Integrated Infrastructure White Paper, September 23, 2014.

## Overview

This document describes the requirements and the analysis model for the Access Control Subsystem of the Predictive Wastewater Management System (PWMS). It provides a secure interface for the Data Analyst to develop predictive algorithms (in addition to others) using data stored into the Data Storage PWMS subsystem.

# Current System

There is no current system in place for the PWMS. However, there are a number of applications involving participatory sensing systems and how those systems operate and can be made secure. The literature shall be thoroughly researched to leverage the state-of-the-art in implementing and securing the PWMS from unauthorized users.

# Proposed System

Our vision closely resembles the system described in the Black and Veatch Whitepaper. By employing the scenario/ use case driven approach we intend to define how end users will employ the system toward the measurement, monitoring, analysis, storage and role-enforcement functions of the PWMS.

## Overview

The Data Analysis subsystem provides a means for the Data Analyst to create Predictive (and other) algorithms using the data stored from the flowcells in a secure manner. A framework of interfaces to output devices and the other PWMS subsystems is provided in addition to a number of Entity objects shown to be useful in the development of optimization algorithms.

A created algorithm may interface with the Access Control PWMS subsystem to enforce authentication and authorization for users of that algorithm.

## Functional Requirements

1. The DAS shall allow the Data Analyst to create algorithms for the PWMS.
2. The DAS shall allow the Data Analyst to execute algorithms for the PWMS.
3. The DAS shall authenticate and authorize Data Analysts for the PWMS.
4. The DAS shall provide a means of securing algorithms created by the Data Analysis Subsystem.
5. The DAS shall report Auditable Events to the Audit and Alert Subsystem (AAS).
6. The DAS shall report Alertable Events to the Audit and Alert Subsystem (AAS).
7. The DAS can be configured by the System Administrator.
8. Auditable events are,
   1. Data Analyst logs into the system
   2. Data Analyst logs off of the system
   3. An Algorithm has been created or updated
   4. Configuration file updated.
   5. An Algorithm has been executed.
   6. Any Alertable event
9. Alertable events are
   1. Internal Subsystem Error.
   2. Intrusion Attempt

## Nonfunctional Requirements

Nonfunctional requirements are those requirements that are not found by doing use cases or modeling but are nevertheless required for successful operation of the PWMS.

### Security

Data transmissions between this and other PWMS subsystems shall be secured/encrypted.

## Scenarios

Scenarios are provided for those use cases that have human actors, with the exception of the System Administrator configuring a PWMS subsystem (i.e. modifying the contents of the configuration file). Scenarios are useful in use case development.

### Arun Creates an Algorithm

Arun logs into the Data Analysis subsystem providing his user name and password. The system responds by allowing access to the classes comprising the Data Analysis subsystem. He creates an algorithm that reads data provided by the Data Storage subsystem, performs some transformation on that data and displays data on the Data Storage output display. In certain instances alerts are sent to the Audit and Alert subsystem.

Upon successful testing the algorithm is saved for later retrieval and execution.

### Sirisha Executes an Algorithm

Sirisha selects an algorithm and begins its execution. A request for username and password appears on the Data Analysis output display. Sirisha enters her credentials and she is authenticated and authorized to run the algorithm. Graphical and/or tabular display are shown on the Data Analysis output display. Under certain conditions an alert is annunciated on the Data Analysis output display and sent to the Audit and Alert subsystem.

## Use Cases

### Administer DAS

|  |  |  |  |
| --- | --- | --- | --- |
| **PWMS Use Case** | | | |
| ***Project*** | Administer Data Analysis | | |
| ***ID Number*** | DAS001 | ***Status*** | For Customer Review |
| ***Creation Date*** | 2/28/15 | ***Last Revision Date*** | 4/5/15 |
| ***Author(s)*** | W. Phillips | | |
| ***Requirements Map*** | 7,8,5 | | |

**Description/Intent**

**T**his Use Case describes the System Administrator administering the Data Analysis Subsystem.

**Actors**

System Administrator, Audit and Alert Subsystem

**Extends**

None

**Uses**

None

**Pre-Conditions**

The Data Analysis Subsystem is operating normally.

**Post-Conditions**

The Data Analysis Subsystem is operating normally. The Data Analysis subsystem configuration is updated.

**Ideal Course of Action**

1. System Administrator logs into the Data Analysis subsystem.
2. System Administrator selects the “Configure Data Analysis Subsystem” on the System Admin dashboard.
3. A menu of configurable data is displayed.
4. Changes are made to the configurable data.
5. System Administrator selects “configure”
6. An auditable event is sent to the Audit and Alert subsystem
7. The use case ends.

**Exceptional Course of Action**

None

**Comments**

None

### Authenticate and Authorize Data Analyst

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PWMS Use Case** | | | | |
| ***Project*** | Authenticate and Authorize Data Analyst | | | |
| ***ID Number*** | DAS002 | ***Status*** | For Customer Review | |
| ***Creation Date*** | 2/28/15 | ***Last Revision Date*** | | 4/5/15 |
| ***Author(s)*** | W. Phillips | | | |
| ***Requirements Map*** | 3,8,5 | | | |

**Description/Intent**

**T**his use case describes a Data Analyst logging in to the Data Analysis subsystem.

**Actors**

Data Analyst, Audit and Alert Subsystem

**Extends**

None

**Uses**

None

**Pre-Conditions**

The Data Analysis Subsystem is operating normally.

**Post-Conditions**

The Data Analysis Subsystem is operating normally.

**Ideal Course of Action**

1. Data Analyst enters his credentials into the Data Analysis Subsystem.
2. Credentials are sent to the ACS
3. Data Analysis enters or is otherwise given access to the Data Analysis Development Environment.
4. An auditable event is sent to the Audit and Alert Subsystem.

**Exceptional Course of Action**

1. Data Analyst enters bad username or password.
2. Error message is displayed and use case returns to step 1.

**Comments**

none

### Data Analyst Creates Algorithm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PWMS Use Case** | | | | |
| ***Project*** | Data Analyst Creates Algorithm | | | |
| ***ID Number*** | DAS003 | ***Status*** | For Customer Review | |
| ***Creation Date*** | 2/28/15 | ***Last Revision Date*** | | 4/5/15 |
| ***Author(s)*** | W. Phillips | | | |
| ***Requirements Map*** | 1,4,5,8 | | | |

**Description/Intent**

This use case depicts a Data Analyst creating an algorithm.

**Actors**

Data Analyst, Data Storage Subsystem, Audit and Alert Subsystem

**Extends**

None

**Uses**

None

**Pre-Conditions**

The Data Analysis Subsystem is operating normally, Data Analyst successfully logged in.

**Post-Conditions**

The Data Analysis Subsystem is operating normally. A new algorithms is catalogued as a runnable algorithm.

**Ideal Course of Action**

1. Data Analyst uses the Interactive Development Environment of the DataAnalystInterface to create an algorithm.
2. Data is requested and received from the Data Storage Subsystem.
3. Data Analyst debugs algorithm and identifies it as completed.
4. An auditable event is sent to the Audit and Alert Subsystem.
5. The use case ends.

**Exceptional Course of Action**

1. Data Analyst debugs algorithm but it is not complete.
2. Data Analysts stores the partially completed algorithm.
3. The use case ends.

**Comments**

none

### Data Analyst Executes Algorithm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PWMS Use Case** | | | | |
| ***Project*** | Data Analysis System | | | |
| ***ID Number*** | DAS003 | ***Status*** | For Customer Review | |
| ***Creation Date*** | 3/1/2015 | ***Last Revision Date*** | | 4/5/15 |
| ***Author(s)*** | W. Phillips | | | |
| ***Requirements Map*** | 2,4,5,6,8,9 | | | |

**Description/Intent**

This use case depicts a Data Analyst executing a pre-developed algorithm.

**Actors**

Data Analyst, Audit and Alert Subsystem, Data Storage Subsystem, Access Control Subsystem

**Extends**

None

**Uses**

None

**Pre-Conditions**

The Data Analysis Subsystem is operating normally, Data Analyst successfully logged in.

**Post-Conditions**

The Data Analysis Subsystem is operating normally.

**Ideal Course of Action**

1. Data Analyst invokes the Stored Algorithm Menu
2. Data Analyst selects the Algorithm to be run
3. Data Analysis selects Run
4. Data Analyst is prompted for credentials
5. Data Analyst enters credentials
6. Access control authenticates and authorizes the algorithm
7. Data Analyst is prompted to enter runtime parameters for the algorithm
8. Data Analyst enters runtime parameters (e.g., stopping criteria, etc.)
9. Algorithm begins execution
10. Data is requested and received from the Data Storage subsystem
11. Audit and Alert Subsystem is sent auditable and alertable events
12. Use case ends.

**Exceptional Course of Actions**

1. Access control fails to authenticate and authorize the algorithm
2. Algorithm terminates
3. Use case continues at step 2

**Comments**

None

### Use Case Diagram



## Object Model

The Object Model is represented by the Conceptual Class Diagram shown on Figure 3.6.14. The boundary, entity and control objects are described as follows.

### SAInterface (Boundary)

SystemAdminInterface is used by the System Administrator to configure the Data Analysis subsystem.

### DataAnalystInterface (Boundary)

DataAnalystInterface is used by the Data Analyst to create and execute algorithms.

### ACSDASInterface (Boundary)

This boundary interface is a shared interface with the Access Control Subsystem.

### DASDSSInterface (Boundary)

This boundary interface is a shared interface with the Data Storage Subsystem.

### AASDASInterface (Boundary)

This boundary interface is a shared interface with the Audit and Alert Subsystem.

### FrameworkBoundaryComponents (Boundary)

FrameworkBoundaryComponents are boundary (interface) components that an algorithm is built with. They primarily provide secure interfaces to the other PWMS subsystems.

### SystemAdminControl (Control)

SystemAdminControl controls the interactions within the Data Analysis subsystem to accomplish the System Admin functions (change configuration data).

### DataAnalysisControl (Control)

DataAnalysisControl controls the interactions within the Data Analysis subsystem to accomplish Data Analysis functions (create, debug, store, execute algorithms).

### PredictiveAlgorithm (Control)

PredictiveAlgorithm is an algorithm created by the Data Analyst and can be run standalone. It controls interactions amongst the FrameworkBoundaryComponents and FrameworkEntityComponents tha the predictive algorithm was built with.

### ConfigurationData (Entity)

ConfigurationData is collection of configuration data used by the Data Analysis Subsystem.

### AlgorithmStorage (Entity)

AlgorithmStorage contains algorithms under development and ready for execution.

### AlgorithmFactory (Entity)

AlgorithmFactory creates, configures and emits an executable algorithm from the framework.

### FrameworkEntityComponents (Entity)

FrameworkEntityComponents are entity objects that the PredictiveAlgorithm is built with. They are primarily abstract data type used to store and process data by the PredictiveAlgorithm.

### Conceptual Class Diagram



## User Interface – Screen Mock-Ups

TBD

# Dynamic Model

The dynamic model is comprised of sequence diagrams, communications diagrams and optional state diagrams for interesting objects. The sequence diagrams correspond to each of the use cases presented in this RAD. Communication diagrams convey the same information but show more clearly the messages passes to and from each object in the object model.

## Administer DAS

This section describes the Systems Administrator configuring DSS.

### Sequence Diagram



### Communication Diagram



## Authenticate and Authorize Data Analyst

This section describes the authentication and authorization of the Data Analyst.

### Sequence Diagram



### Communication Diagram



## Data Analyst Creates Algorithm

This section describes the Data Analyst creating an algorithm.

### Sequence Diagram



### Communication Diagram



## Data Analyst Executes Algorithm

This section describes the Data Analyst executing a previously created algorithm.

### Sequence Diagram



### Communication Diagram

