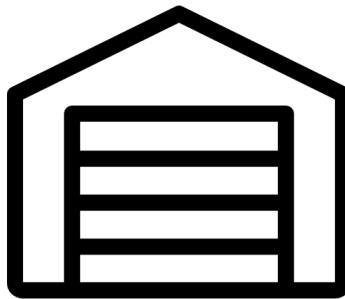


**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**ARCHITECTURAL DESIGN SPECIFICATION
CSE 4316: SENIOR DESIGN I
SPRING 2023**



**TEAM STOCKERS
GENERAL INVENTORY MANAGMENT**

**IVAN CUARENTA
JOSEPH SALAS
JEREMY WALKER
RODNEY RAMIREZ**

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	04.26.2023	IC, JS, JW, RR	document creation
0.2	7.26.2023	JW	Table of interfaces in Front-end and Back-end
0.3	7.31.2023	JW	Database, Encryption, and Reporting and Analytic
0.4	8.1.2023	JW	Amended tables and images for System Overview, Front-end, Back-end, Database, Encryption, Price Prediction, and Reporting & Analytic. Updated image for Data Flow

CONTENTS

1	Introduction	5
2	System Overview	6
2.1	Front-End Layer	6
2.2	Back-End Layer	6
2.3	Database Layer	6
2.4	Encryption Layer	6
2.5	Prediction Layer	6
2.6	Reporting & Analytic Layer	7
3	Subsystem Definitions & Data Flow	8
4	Front-End Layer Subsystems	9
4.1	Landing Page	9
4.2	Log In	10
4.3	Registration	10
4.4	Dashboard	11
5	Back-End Layer Subsystems	13
5.1	Adding/Updating Asset	13
5.2	Removing Asset	14
5.3	Generating Report	14
5.4	Checkout Asset	15
5.5	Prediction	16
5.6	Log out	16
6	Database Layer Subsystems	17
6.1	Company	17
6.2	User	18
6.3	Inventory Table	19
7	Encryption Layer Subsystems	20
7.1	Encryption	20
7.2	Decryption	21
8	Price Prediction Layer Subsystems	22
8.1	Price prediction	22
9	Reporting & Analytic Layer Subsystems	24
9.1	Format Report	24
9.2	Export File	24

LIST OF FIGURES

1	A simple architectural layer diagram	6
2	A simple data flow diagram	8
3	Front End Layer	9
4	Back-End Layer	13
5	Database	17
6	Encryption	21
7	Price Prediction	22
8	Example subsystem description diagram	24

LIST OF TABLES

2	Landing Page interfaces	9
3	Log In interfaces	10
4	Registration interfaces	11
5	Dashboard interfaces	12
6	Adding/Updating Asset interface	14
7	Removing Asset interface	14
8	Generating Report interface	15
9	Checkout Asset interface	15
10	Prediction interface	16
11	Logout interface	16
12	Company interfaces	18
13	User interfaces	18
14	Inventory table interfaces	20
15	Encrypton interfaces	21
16	Decryption interfaces	22
17	Subsystem interfaces	23
18	Format Report Interfaces	24
19	Export Interfaces	25

1 INTRODUCTION

This project is an inventory management program that aids a person or entity to maintain an accurate inventory list. The project is designed to be as streamlined as possible to ensure that the user is able to enter an asset's attributes, such as serial numbers, descriptions, and images, with ease. There are several requirements that must be met in order to ensure that the program has a strong architectural design. The first requirement is that the user registers/login and so the user sees their own inventory and not someone else's inventory. Secondly, the user can issue an account recovery if the password is compromised or the user wishes to change the password, afterwards the user will be asked to verify the account again. Additionally, when the user enters the asset's attributes they must be listed together and not become entangled with another asset. Furthermore, the user will be able to edit, add, and remove an asset's attributes, add another asset, or remove an asset respectively. Lastly, the account information will be stored in MongoDB, this information is not encrypted so a layer is added to encrypt the data in the event that a malicious actor is given unrestricted access.

2 SYSTEM OVERVIEW

The inventory management system will contain five layers that will communicate with each other to ensure that the key requirements are functioning for the user. The front-end layer ensures that a connection is established between the user and the program. The front end has a sub-layer called the landing page which allows the user to register and log in, then the login information is sent and stored in a database where the password will become encrypted. Then the user will be directed to the primary page where they are able to add, edit, or remove assets. After an asset has been added the encryption layer encrypts a few of the assets such as serial numbers and pictures.

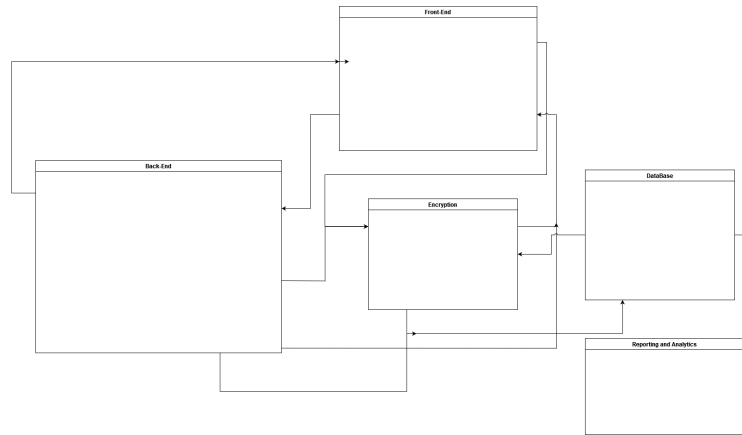


Figure 1: A simple architectural layer diagram

2.1 FRONT-END LAYER

The Front-End layer will be responsible to create a connection between the user and the application itself. This will be done by providing the user with a simple interface that the user will be able to perform their adding, editing, or removal of their assets. After the user passes the login page they will see a table where they will be able to enter a new asset, edit an existing one, or remove an asset.

2.2 BACK-END LAYER

The back-end layer is responsible for communicating with other subsystems in the application. It will handle all the requests from the user to log in or out of the application as well as generate reports, add assets, remove assets, check out an asset, and price predictions.

2.3 DATABASE LAYER

The Database Layer receives all queries from all the other layers that access it and then updates its tables based on the query.

2.4 ENCRYPTION LAYER

The Encryption Layer will take in requests from the Front-end and encrypt specific items that would require encryption such as passwords and serial numbers. When the Front-end queries data from the database it will also pass through the encryption layer so that it can be decrypted and read as plain text for the user.

2.5 PREDICTION LAYER

This layer will be responsible for predicting the value of assets using a machine learning algorithm.

2.6 REPORTING & ANALYTIC LAYER

The Reporting & Analytic Layer will use the data that the user requested and then use a specified format to generate a report. Once the report is formatted the layer will export the file and download it to the host system

3 SUBSYSTEM DEFINITIONS & DATA FLOW

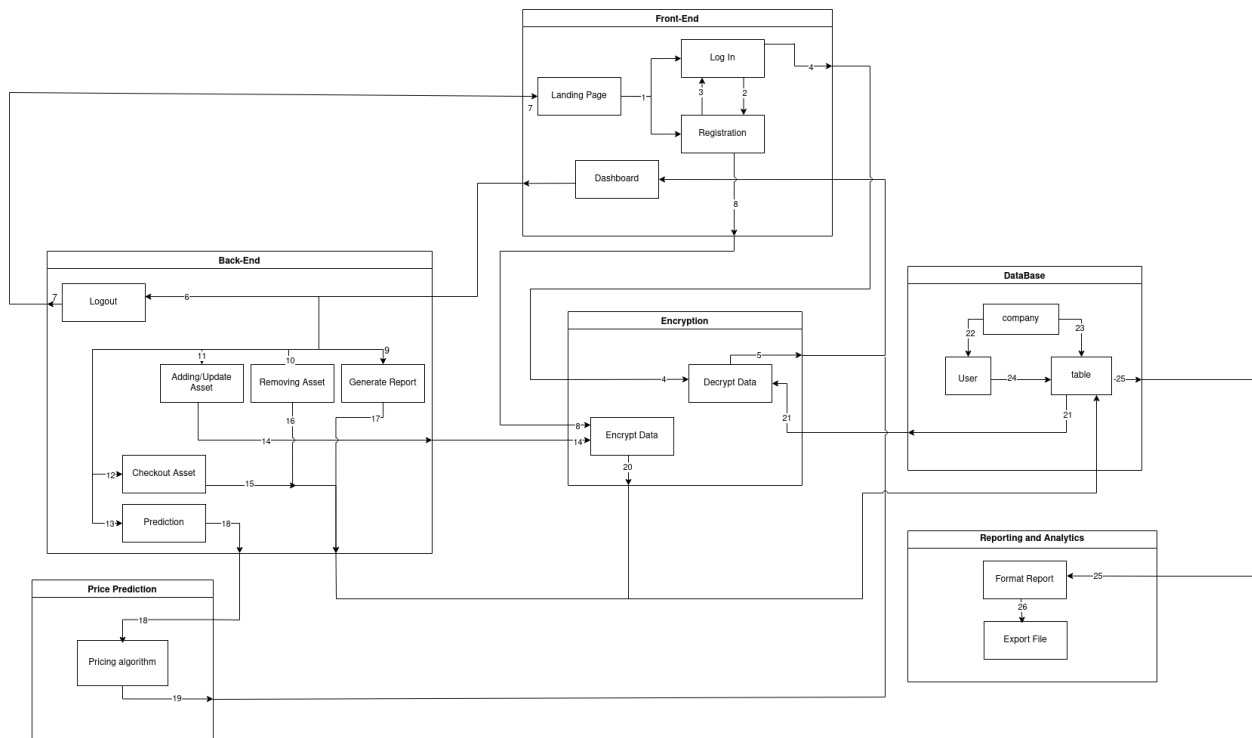


Figure 2: A simple data flow diagram

4 FRONT-END LAYER SUBSYSTEMS

The Front-End layer begins with the Landing Page and depending on the users choice, the landing page will direct them to the Login or Registration page. From there the as long as there are no errors the user will then reach the Dashboard where the user can interact with the interface to access the back-end functionality.

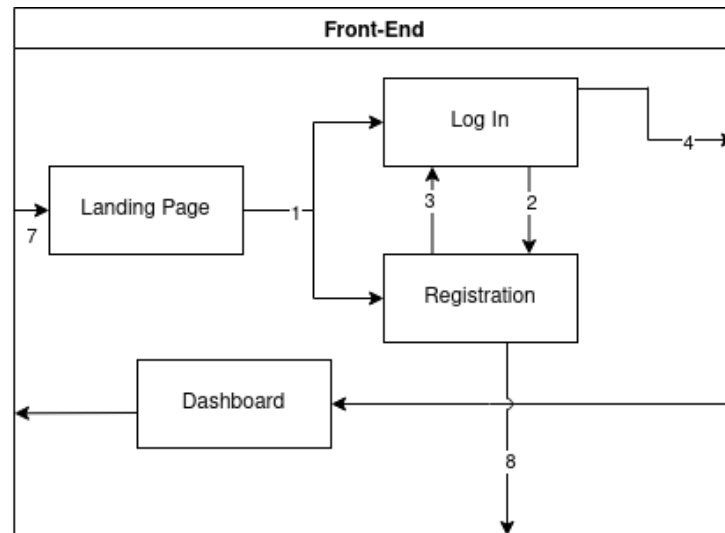


Figure 3: Front End Layer

4.1 LANDING PAGE

On the Landing Page the user can view the many functions that the Web App can exhibit. From here the user can choose to click on Register or Login buttons. That will direct them to their respective pages.

4.1.1 ASSUMPTIONS

- This is the first page a user will see

4.1.2 RESPONSIBILITIES

- Explain the features of the Web App
- Take user to the Login page
- Take user to the Registration page

4.1.3 LANDING PAGE INTERFACES

Table 2: Landing Page interfaces

ID	Description	Inputs	Outputs
#1	Login Button	Login request	N/A
#1	Registration Button	Create new account request	N/A
#7	Logout	N/A	Logout request

4.2 LOG IN

The Login page will take the users E-mail and Password and send it out to be verified. If the user wants to register instead, they can be redirected to the Register page.

4.2.1 ASSUMPTIONS

- The user already has an account

4.2.2 RESPONSIBILITIES

- Receive the users credentials
- Take user to the Registration page

4.2.3 LOG IN INTERFACES

Table 3: Log In interfaces

ID	Description	Inputs	Outputs
#1	Landing Page	e-mail password login request	N/A
#2	Registration	N/A	Registration re- quest
#3	Registration	Registration re- quest	N/A
#4	Decrypt Data	N/A	Users password to compared with stored password

4.3 REGISTRATION

The Registration page will take the user's E-mail and Password and send it out to be verified and added to the database. Users can be redirected to the Login page if they want to log in instead.

4.3.1 ASSUMPTIONS

- The user wants to create an account

4.3.2 RESPONSIBILITIES

- Receive the user's credentials
- Take the user to the Login page

4.3.3 REGISTRATION INTERFACES

Table 4: Registration interfaces

ID	Description	Inputs	Outputs
#1	Landing page	e-mail password registration re- quest	N/A
#2	Login	N/A	Login request
#3	Login	Login request	N/A
#8	Encrypt Data	e-mail password button press	User's encrypted password

4.4 DASHBOARD

The Dashboard will be a hub of all the functionalities of the Web App. This is where users will send requests and information to the back end to be processed.

4.4.1 ASSUMPTIONS

- The user has an account
- The is logged in

4.4.2 RESPONSIBILITIES

- Display specialized UI for user
- Provide buttons for each function
- Easy to Navigate

4.4.3 DASHBOARD INTERFACES

Table 5: Dashboard interfaces

ID	Description	Inputs	Outputs
#5	Decrypt Data	Decrypted data from the inventory table	N/A
#6	Logout Button	N/A	Log out request
#11	Add Asset	N/A	asset attributes
#10	Remove Asset	N/A	asset id Remove request
#12	Checkout Asset	N/A	asset id checkout request
#9	Generate Report	N/A	asset id Report request
#19	Pricing Alogrithim	N/A	price id price re- quest

5 BACK-END LAYER SUBSYSTEMS

In this section, the layer consists of the following subsystems adding/updating assets, removing assets, generating a report, checking out assets, predicting, and finally logout.

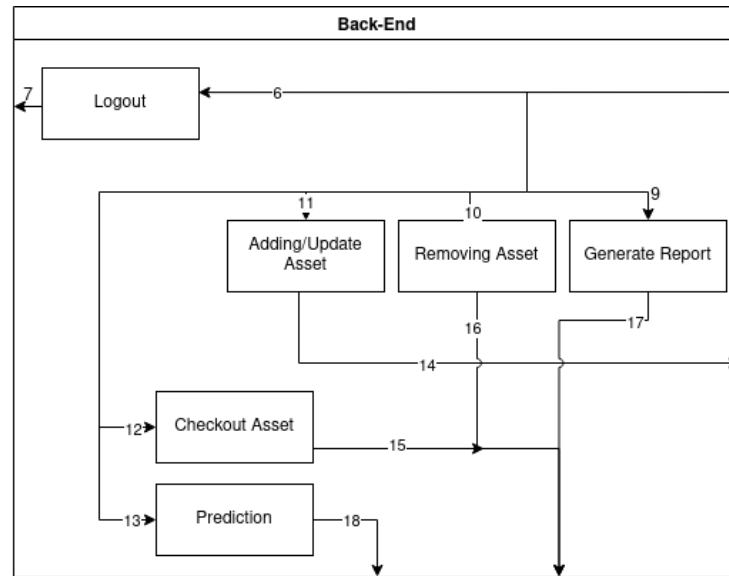


Figure 4: Back-End Layer

5.1 ADDING/UPDATING ASSET

In this subsystem, assets are added and/or updated in the inventory system.

5.1.1 ASSUMPTIONS

- There is enough memory in the database
- User has access to the correct table
- Serial numbers are known
- Image of asset is provided

5.1.2 RESPONSIBILITIES

- New asset(s) are added to the tables stored in the database
- Existing asset(s) are updated with inventory count
- Serial number and image are encrypted
- Return error is asset does not exist

5.1.3 ADDING/UPDATING ASSET INTERFACE

Table 6: Adding/Updating Asset interface

ID	Description	Inputs	Outputs
# 11	Dashboard	Fields for asset name asset description asset serial number asset images	N/A
# 14	Encrypt Data	N/A	Asset Quantity Serial number Image Item Description

5.2 REMOVING ASSET

Assets are removed from the inventory system and the count is updated.

5.2.1 ASSUMPTIONS

- Valid asset/quantity is sent to the subsystem

5.2.2 RESPONSIBILITIES

- Verify asset/quantity is in the database, else return error
- Remove asset and update count

5.2.3 REMOVING ASSET INTERFACE

Table 7: Removing Asset interface

ID	Description	Inputs	Outputs
#10	Dashboard	A list of assets the user wishes to remove	N/A
#16	Table	N/A	The list of unique ids to be removed from the table.

5.3 GENERATING REPORT

A general report will be generated consisting of an image, serial number and count of the specified asset(s).

5.3.1 ASSUMPTIONS

- The asset(s) requested by the user is in the database. Which includes the inventory count, serial number, and image.

5.3.2 RESPONSIBILITIES

- Report is generated from using information stored in database

5.3.3 GENERATING REPORT INTERFACE

Table 8: Generating Report interface

ID	Description	Inputs	Outputs
# 9	Dashboard	A list of assets that the user can choose to run a report on	N/A
#17	Table	N/A	PDF of the report

5.4 CHECKOUT ASSET

The encrypted information of the asset will be decrypted and available for the user to view.

5.4.1 ASSUMPTIONS

- Asset information is available
- Serial number and image are encrypted

5.4.2 RESPONSIBILITIES

- Verify the user has access to tables
- Verify asset exists in the database
- Asset's attributes (serial number and image) are decrypted

5.4.3 CHECKOUT ASSET INTERFACE

Table 9: Checkout Asset interface

ID	Description	Inputs	Outputs
# 12	Dashboard	The user requests to check out an asset or assets	N/A
# 17	Table	N/A	The user requests to check out an asset or assets

5.5 PREDICTION

The price for an asset will be predicted using the price predicting algorithm in the event the user or company's asset is involved in theft or damage.

5.5.1 ASSUMPTIONS

- Pricing algorithm is accurate on asset price
- Asset prices are available

5.5.2 RESPONSIBILITIES

- Generate price for assets

5.5.3 PREDICTION INTERFACE

Table 10: Prediction interface

ID	Description	Inputs	Outputs
# 13	Dashboard	The requested asset ID	N/A
# 18	Pricing Algorithm	N/A	The asset ID and name.

5.6 LOG OUT

The user will be logged out and directed back to the landing page of the inventory system.

5.6.1 ASSUMPTIONS

- User is logged in with the correct credentials
- User is ready to log out of the system

5.6.2 RESPONSIBILITIES

- Log the user out of the inventory system
- Return to the landing page

5.6.3 LOGOUT INTERFACE

Table 11: Logout interface

ID	Description	Inputs	Outputs
# 6	Dashboard	User requested to log out	N/A
# 7	Landing Page	N/A	Prompt for user to log in

6 DATABASE LAYER SUBSYSTEMS

The database layer will use the MongoDB system to store all data for all the customers. It will hold company information as well as users and their information, and inventory tables with information that is applicable to the inventory input by the company.

6.1 COMPANY

The company subsystem will hold the login ID and passwords for companies who sign up to use our system. Each company will have users and inventory tables. The company will have a company ID.

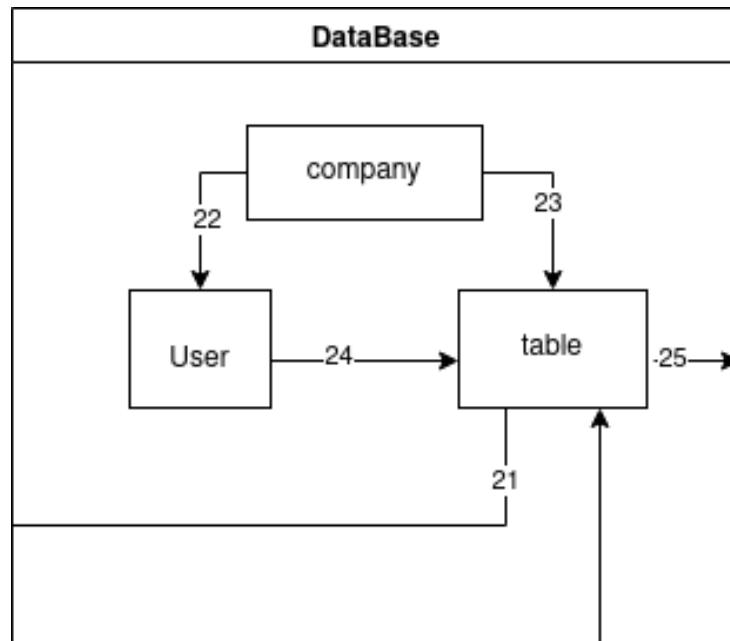


Figure 5: Database

6.1.1 ASSUMPTIONS

- The company account password will be encrypted.
- The Company ID will be unique.
- The company can create as many users as it needs.
- The company can create as many inventory tables as it needs.
- The company will be able to make as many users and admins as it needs.

6.1.2 RESPONSIBILITIES

- Retrieving and decrypting the password upon entry of the correct password
- Giving the company access to all users created under that company.
- Giving the company access to all inventory tables that are associated with that company and its ID.

6.1.3 COMPANY INTERFACE

Table 12: Company interfaces

ID	Description	Inputs	Outputs
#23	User	N/A	user account profile
#24	Inventory table	N/A	A schema with predefined attributes.

6.2 USER

There is a subsystem for the user accounts created by the companies. The user's email will be stored in the database as their login ID and they will have a password of their choosing. The user will be assigned a group number by the admin. The user will have a field for the company ID associated with them. The user will also have an admin attribute of type Boolean.

6.2.1 ASSUMPTIONS

- Users will only have access to the tables that are associated with their company ID and group number.
- Admins will have access to all tables associated with their company ID.
- Admins can assign group numbers to other users.

6.2.2 RESPONSIBILITIES

- Allowing users to update inventory on tables they have access to.
- Allowing admins to run reports on the inventory.
- Allowing users to run reports on the inventory.
- Ensuring the data cannot be accessed by unauthorized means.

6.2.3 USER INTERFACES

Table 13: User interfaces

ID	Description	Inputs	Outputs
#23	Company	user ID admin status	N/A
#25	Inventory Table	N/A	Data requested or confirmation of data added

6.3 INVENTORY TABLE

The subsystem for inventory tables will hold all the tables the company sets up for its assets. Each table will have a company ID attribute associated with it as well as a group number. The table will hold item names, descriptions of the item, a picture of the item, category, subcategory, price, and serial number if applicable

6.3.1 ASSUMPTIONS

- The tables will be up to date with the current inventory.
- The tables will have the correct company ID associated with them.
- Each table will have the correct group ID.
- The tables will only be accessible by authorized users.

6.3.2 RESPONSIBILITIES

- Keep all data given to it.
- Encrypt the data until it is queried and sent to the requesting authorized user.
- Enforce data types of each attribute.

6.3.3 INVENTORY TABLE INTERFACES

Table 14: Inventory table interfaces

ID	Description	Inputs	Outputs
#15	Checkout Asset	The id of the asset requested for checkout	N/A
#16	Removing Asset	The id of the asset to be removed	N/A
#17	Generate Report	A list of item ids that a report is being requested for report	N/A
#20	Encrypt Data	Encrypted data to be stored in the appropriate attributes	N/A
#21	Decrypt Data	N/A	Encrypted data from the inventory table
#23	Company	create inventory table	N/A
#24	User	Request asset data	N/A
#25	Format Report	N/A	The assets and all information associated with each asset that was requested for the report

7 ENCRYPTION LAYER SUBSYSTEMS

The encryption layer will handle encryption for sensitive information such as passwords, serial numbers, and images.

7.1 ENCRYPTION

The encrypt data subsystem will simply encrypt the data as needed or requested by the customer. Encrypting serial numbers and images will be optional. Password encryption will be mandatory.

7.1.1 ASSUMPTIONS

- The password encryption will be strong enough that it would take too long to break before the password is reset.
- The same algorithm can be used to encrypt other data of the same type.
- Encrypted data will be useless if it is taken by an unauthorized party.

7.1.2 RESPONSIBILITIES

- Encrypt data on the client side.
- Keep data encrypted while on the server.

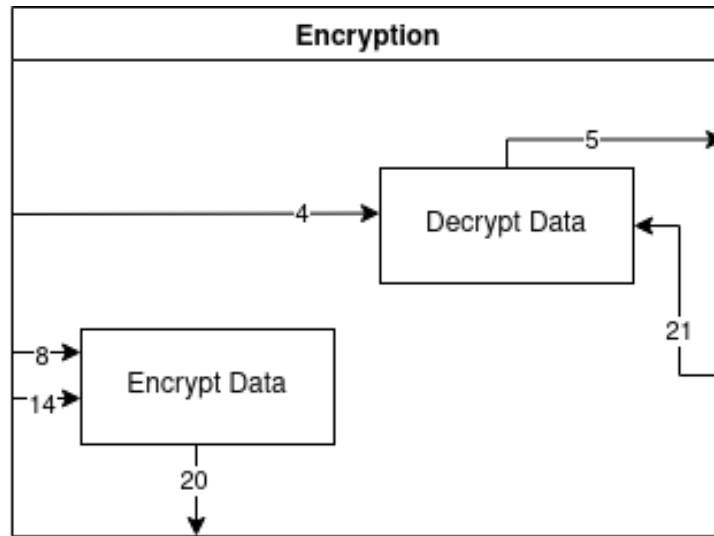


Figure 6: Encryption

7.1.3 ENCRYPTION INTERFACES

Table 15: Encrypton interfaces

ID	Description	Inputs	Outputs
#8	Registration	The users pass-word	N/A
#14	Adding / Updating Asset	The values from asset entry that are to be encrypted.	N/A
#20	Database	N/A	Encrypted data

7.2 DECRYPTION

The decrypt data layer decrypts the data when requested by authorized users.

7.2.1 ASSUMPTIONS

- The data will only be decrypted for a user once their username and password have been verified.
- The same algorithm can be used to decrypt other data of the same type.

7.2.2 RESPONSIBILITIES

- Decrypt data on the client side.

7.2.3 SUBSYSTEM INTERFACES

Table 16: Decryption interfaces

ID	Description	Inputs	Outputs
#4	Login	The users password	N/A
#5	Dashboard	N/A	The inventory table that is associated with the user is displayed
#21	Database	The encrypted data from the inventory table	N/A

8 PRICE PREDICTION LAYER SUBSYSTEMS

The prediction layer will handle the price prediction of items in the event that assets are lost or stolen.

8.1 PRICE PREDICTION

The prediction subsystem will display an asset's predicted price in the event that it's lost or stolen.

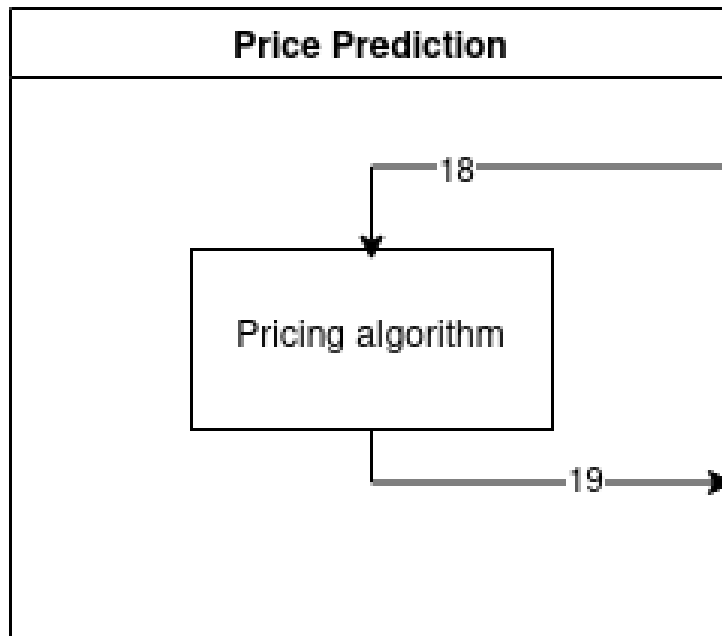


Figure 7: Price Prediction

8.1.1 ASSUMPTIONS

- After the user enters an asset a price value is generated.

8.1.2 RESPONSIBILITIES

- Using a machine learning algorithm the prediction layer should generate an accurate price value of the asset.

8.1.3 SUBSYSTEM INTERFACES

Table 17: Subsystem interfaces

ID	Description	Inputs	Outputs
#18	Prediction	Asset details and specifications.	N/A
#19	Dashboard	N/A	A price based on the current market value of similar items.

9 REPORTING & ANALYTIC LAYER SUBSYSTEMS

This Layer generates a report and formats the report using pre-defined templates. Once the report has been generated and formatted the Export File will download the file to the system.

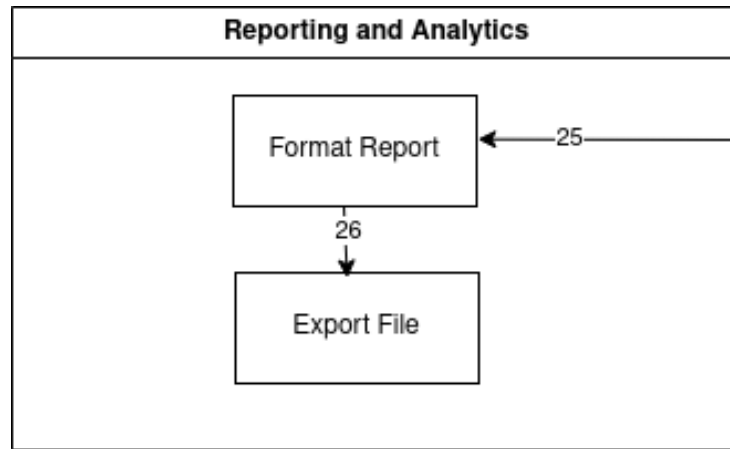


Figure 8: Example subsystem description diagram

9.1 FORMAT REPORT

The Format Report subsystem will take the file type and report template received from the back end and the actual asset information from the database and format the asset information to the decided template.

9.1.1 ASSUMPTIONS

- At least one asset has been selected

9.1.2 RESPONSIBILITIES

- Format asset information to the selected template and file type

9.1.3 FORMAT REPORT INTERFACES

Table 18: Format Report Interfaces

ID	Description	Inputs	Outputs
#25	Table	file type asset list template	N/A
#26	Export File	N/A	The PDF file for the user to down- load

9.2 EXPORT FILE

This subsystem takes the formatted file and downloads the file to the host system.

9.2.1 ASSUMPTIONS

- File is formatted correctly

9.2.2 RESPONSIBILITIES

- Take a formatted file and download the file to the system

9.2.3 EXPORT INTERFACES

Table 19: Export Interfaces

ID	Description	Inputs	Outputs
#26	Formatted Report	formatted file	N/A

REFERENCES