



## **Institute of Information Technology University of Dhaka**

**Project name: Layout-Congestion Management System (LCM)**

**Supervised by:**  
Kishan Kumar Ganguly  
Associate Professor

**Submitted by:**  
Abhijit Paul (1201)  
Sharif Mohammad Abdullah (1211)  
Rifah Tashfiha Faria (1213)  
Muktadul Islam (1215)  
Rufidatul Radium (1226)  
Md. Jubaer Hossain (1255)

# Table of Contents

|  |           |
|--|-----------|
| <b>Table of Contents.....</b>                          | <b>2</b>  |
| <b>0.1 Quality Function Deployment.....</b>            | <b>5</b>  |
| Normal requirements.....                               | 5         |
| Expected Requirements.....                             | 5         |
| Exciting Requirements.....                             | 6         |
| <b>0.2 Major Tasks.....</b>                            | <b>6</b>  |
| <b>0.3 Sequence Diagram.....</b>                       | <b>7</b>  |
| Login & Line Creation.....                             | 8         |
| Change Layout For Line Balancing.....                  | 9         |
| Report Generation.....                                 | 9         |
| <b>1. Architectural Context Diagram.....</b>           | <b>10</b> |
| <b>2. Defining Architectural Patterns.....</b>         | <b>11</b> |
| 2.1 Layered Architecture.....                          | 11        |
| 2.2 Pipe & Filter Architecture.....                    | 12        |
| 2.4 MVC Architecture.....                              | 12        |
| <b>3. Top Level Components.....</b>                    | <b>14</b> |
| Component Details.....                                 | 14        |
| <b>4. Instantiate Pattern.....</b>                     | <b>15</b> |
| 4.1 Component : Congestion Detection.....              | 15        |
| 4.2 Component : Report Generation.....                 | 16        |
| 4.3 Component : Congestion Issue.....                  | 16        |
| 4.4 Component : Line Metrics.....                      | 17        |
| 4.5 Component : Authorization & Authentication.....    | 18        |
| <b>5. From Analysis Classes to Design Classes.....</b> | <b>18</b> |
| 5.1 Design Class for Line Metrics Component.....       | 19        |
| 5.1.1 Line Layout Formation.....                       | 19        |
| 5.1.2 Line Layout Parser.....                          | 20        |
| 5.1.3 Garment Design Parser.....                       | 21        |
| 5.1.4 Hourly Production Heuristic.....                 | 22        |
| 5.1.5 Layout Efficiency Calculation.....               | 23        |
| 5.2 Design Class for Congestion.....                   | 24        |
| 5.2.1 Congestion Detection.....                        | 24        |
| 5.2.2 Issue Detection.....                             | 25        |
| 5.2.3 Issue Controller.....                            | 26        |

|   |           |
|---|-----------|
| <b>6. Message Passing between Classes.....</b>            | <b>27</b> |
| 6.1 Line Metrics Component.....                           | 27        |
| 6.2 Congestion Component.....                             | 28        |
| <b>7. Elaborate attributes and define data types.....</b> | <b>29</b> |
| 7.1 Line Metrics Component.....                           | 29        |
| 7.2 Congestion Component.....                             | 31        |
| <b>8. Process Flow and Pseudocodes.....</b>               | <b>34</b> |
| 8.1 Congestion Component.....                             | 34        |
| 8.1.1 mark_workstation().....                             | 34        |
| 8.1.2 resource_allocation().....                          | 35        |
| 8.1.3 mechanic_availability().....                        | 36        |
| 8.1.4 record_issue().....                                 | 37        |
| 8.1.5 set_congestion_reason().....                        | 38        |
| 8.2 Congestion Component - ML.....                        | 38        |
| 8.2.1 Class: Configuration Singleton.....                 | 38        |
| 8.2.2 Class: Preprocess Image.....                        | 39        |
| <b>9. Behavioral Representations for Classes.....</b>     | <b>41</b> |
| 9.1 Line Metrics Component.....                           | 41        |
| 9.1.1 AssemblyLine.....                                   | 41        |
| 9.1.2 LayoutDesigner.....                                 | 42        |
| 9.1.3 LayoutConfirmation.....                             | 43        |
| 9.1.4 LayoutParser.....                                   | 44        |
| 9.1.5 LineBalancer.....                                   | 45        |
| 9.1.6 HourlyProductionPrediction.....                     | 46        |
| 9.2 Congestion Component.....                             | 47        |
| 9.2.1 CongestionDetection.....                            | 47        |
| 9.2.2 CongestionDetectionModel.....                       | 48        |
| 9.2.3 IssueDetection.....                                 | 49        |
| 9.2.4 Issue Detection Model.....                          | 50        |
| 9.2.5 Issue Controller.....                               | 51        |
| 10. User Interface Design.....                            | 52        |
| 10.1 User Analysis.....                                   | 52        |
| 10.1.1 Supervisor.....                                    | 52        |
| 10.1.2 Line Chief.....                                    | 53        |
| 10.1.3 Production Manager.....                            | 54        |
| 10.1.4 Mechanic.....                                      | 55        |
| 10.2 Task Analysis.....                                   | 56        |
| 10.2.1 User - Supervisor.....                             | 56        |
| 10.2.2 User - Mechanic.....                               | 57        |
| 10.2.3 User - Line Chief.....                             | 57        |
| 10.2.4 User - Production Manager.....                     | 59        |

|  |    |
|--|----|
| 10.3 Design Choices Derived from Golden Rules..... | 60 |
| 10.4 Prototype.....                                | 61 |
| 10.4.1 Video Demonstration.....                    | 77 |
| 11. Elaborating Deployment Diagram.....            | 78 |
| 11.1 Deployment Diagram.....                       | 78 |
| 11.2 Instance Form of Deployment Diagram.....      | 79 |
| 11.3 Reference.....                                | 79 |

# 0.1 Quality Function Deployment

After talking & negotiating with the stakeholders, we have finalized the following requirements for our system. They are:

## Normal requirements

1. The admin can add users to the system.
2. The user can change their user information.
3. Admin can add viewer information.
4. The users can change their account password.
5. Production manager, line chief, supervisor, mechanic, **quality checker**, admin, viewer, and guest users have their own login system.
6. Production managers can input the amount of resources that is available to him.
7. Production manager can upload product type and the different types of tasks required to make the product as the design file
8. Production manager can create assembly line layout
9. Production manager can assign the line to a line chief
10. Production manager defines: the number of products to be produced, daily line targets, fault tolerance levels.
11. Production manager finalizes the assembly line layout.
12. Production manager is notified on hourly production report, request for resource, congestion information
13. Line chief can request mechanics to fix workstations.
14. Supervisors can view assembly line layouts and product details.
15. Supervisor uploads congestion reason
16. Supervisor uploads backpropagation/faulty product information
17. Supervisor can mark over/under workstations of the line.
18. Supervisors upload tutorials for workstations.
19. Mechanics accept/reject workstation fixing requests and insert workstation information(fixed or not, what type of problem occurred)
20. Mechanics uploads daily workstation details
21. The production manager can download reports.

## Expected Requirements

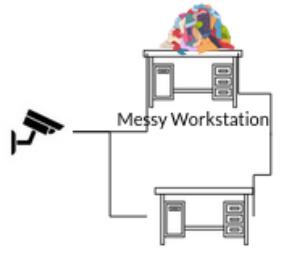
1. Each user will have a different level of access to the system.
2. Notification system to notify about any new reports or messages from other users.
3. Production manager can generate performance matrices on assembly line layout
4. The system must have an admin account.
5. Reports will be digitally generated automatically.
6. Live layout of an assembly line must be updated and viewable at all times

7. Communicate all the reports (production, overtime, washing, sample testing, quality assurance, machine report, congestion report, line balancing report, daily assembly line performance report) to the concerned users

## Exciting Requirements

1. Automated Line balancing based on product
2. Congestion detection in the line based on object/stack detection.
3. Generate layout based on design file and resources

## 0.2 Major Tasks



Congestion Detection



User Interaction



Report Generation

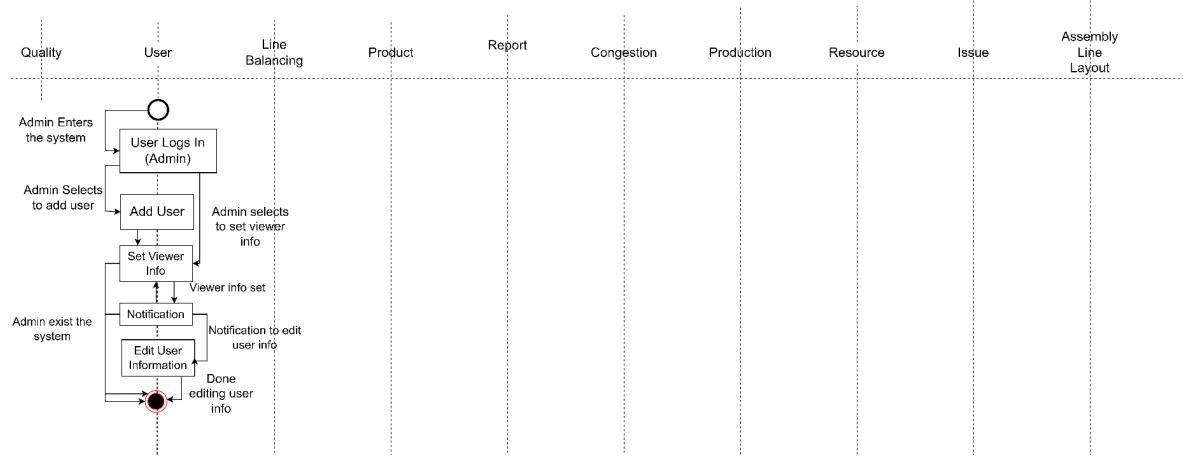
## 0.3 Sequence Diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focused, and they show the order of the interaction visually by using the vertical axis of the diagram to represent time, what messages are sent and when.

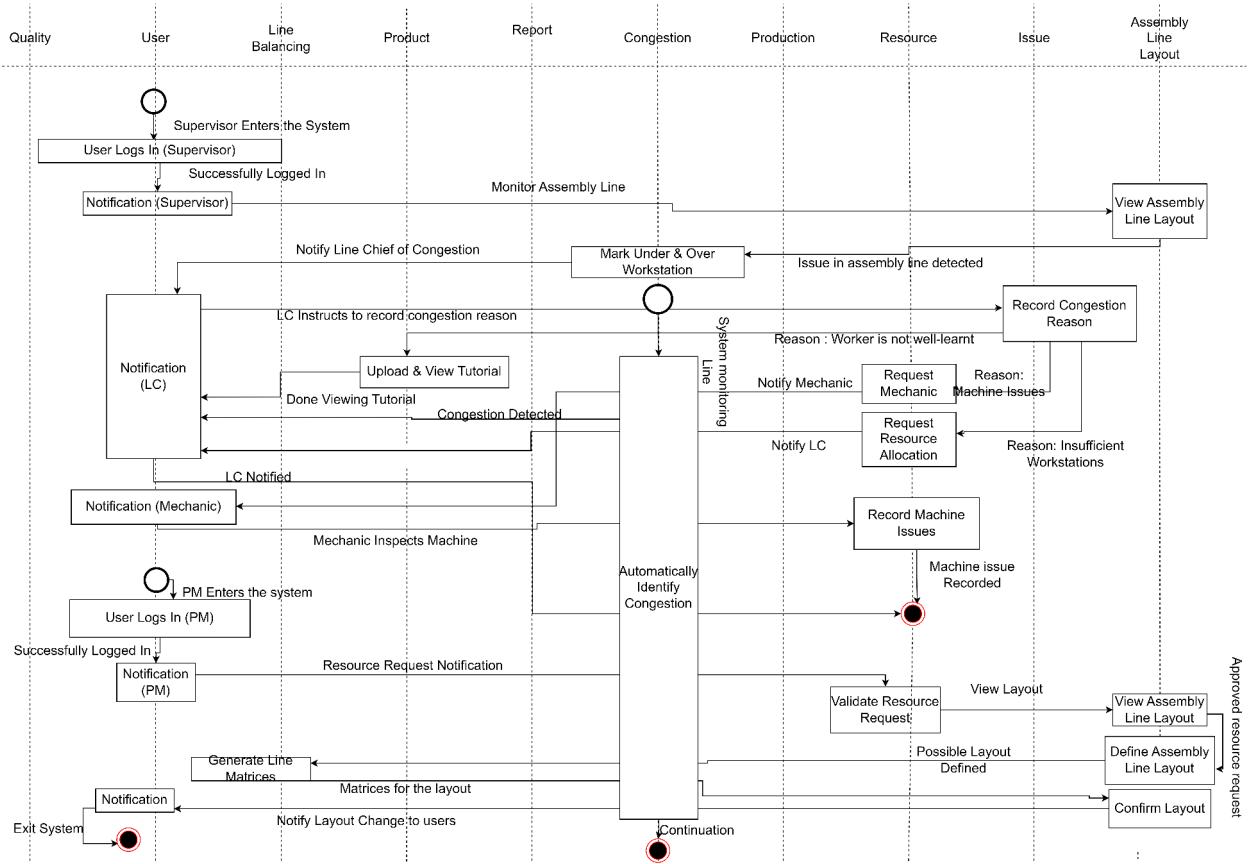
For better quality diagram, enter this link:

<https://app.diagrams.net/#G1JTuyo3z4pbgWdTPiwHHpO2JEkh2IlaGC>

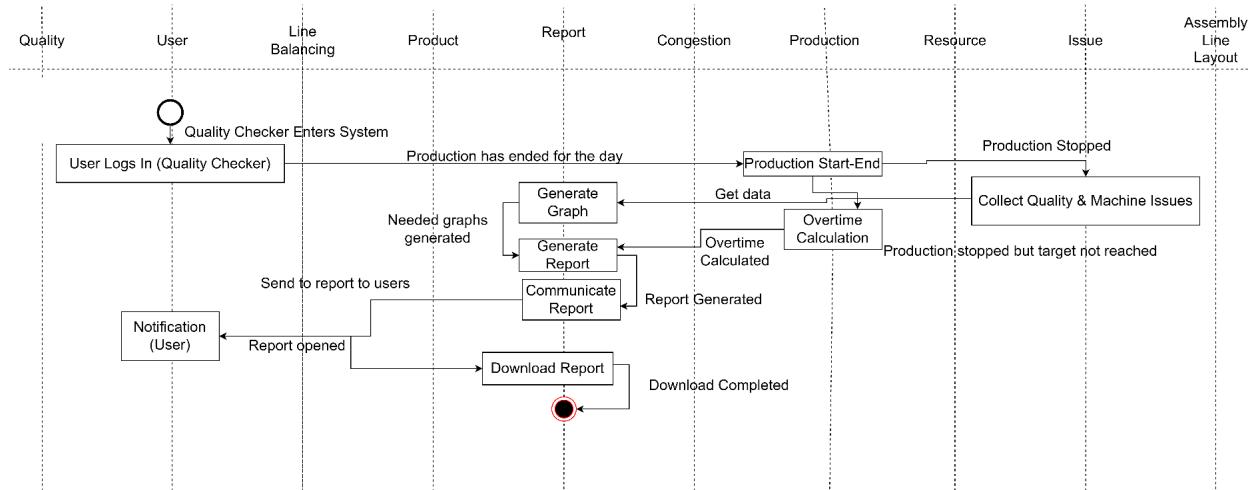
## Login & Line Creation



## Change Layout For Line Balancing



## Report Generation



# 1. Architectural Context Diagram

At the architectural design level, a software architect uses an architectural context diagram (ACD) to model the manner in which software interacts with entities external to its boundaries. The architectural context diagram for LCM is illustrated in Figure-A1.

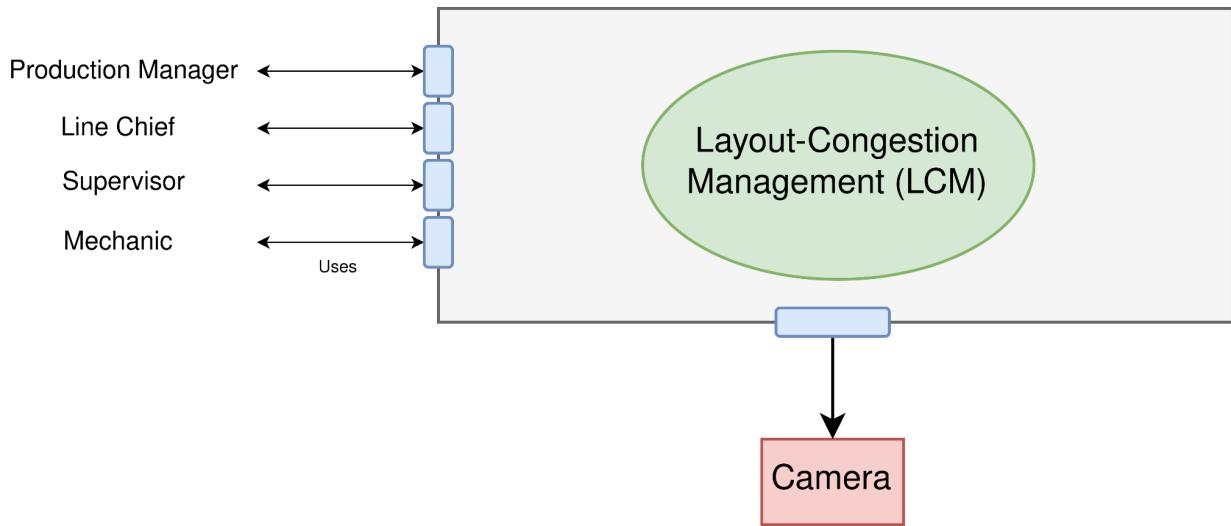


Fig A1 : Architectural Context Diagram of LCM (Layout-Congestion Management)

We have the following subordinate and actors in our system:

1. Camera Sensor
2. Production Manager
3. Line Chief
4. Supervisor
5. Mechanic

The external entity communicates with the LCM system through an interface (the small shaded rectangles).

## 2. Defining Architectural Patterns

Architectural patterns are an integral part of instantiating components. However, the patterns only have generic definitions and don't cover all aspects of diverse real world applications. For this reason, we will start by defining the architectural patterns in our system context.

### 2.1 Layered Architecture

Layered architecture is a software design pattern that organizes a system's components into distinct layers, with each layer serving a specific set of functionalities and having well-defined interfaces for communication. It is commonly used in various applications, from web development to enterprise software, due to its clear separation of concerns and ease of maintenance.

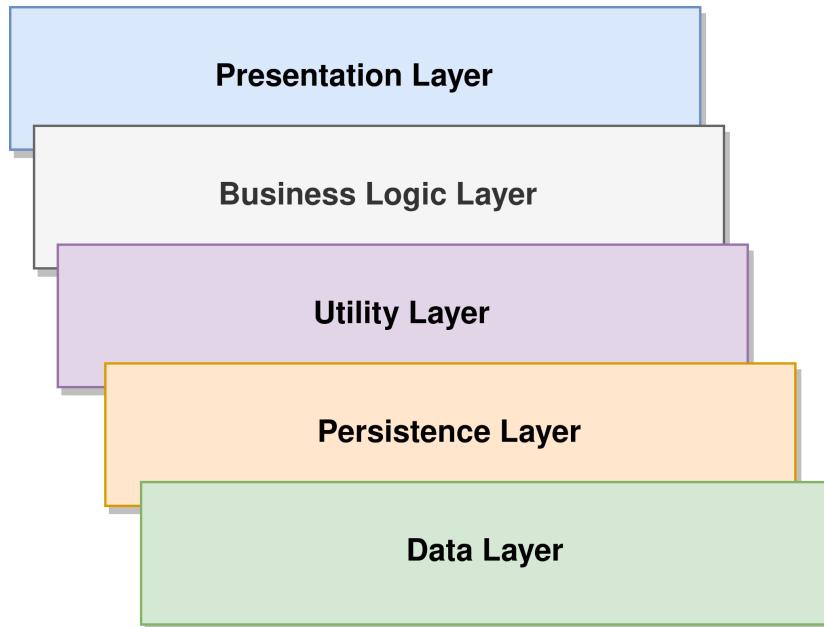


Fig B1: Layered Architecture

1. **Presentation Layer:** This layer gets and presents data from and to the system from the user.
2. **Business Logic Layer:** Contains the main business logic of our system.
3. **Utility Layer:** It contains miscellaneous functions that are used by the business logic layer. It includes library functions, frameworks and well-established algorithms.
4. **Persistence Layer:** It contains interfaces to external entities and document access objects.
5. **Data Layer:** This layer contains the miscellaneous general libraries and modules used by our application. It provides services that the service layer uses.

## 2.2 Pipe & Filter Architecture

The Pipe and Filter architecture is a software design pattern that organizes a system as a series of processing elements (filters) connected by communication channels (pipes). It is particularly useful for designing applications that involve processing streams of data in a linear and sequential manner. Each filter performs a specific task on the incoming data and passes the results to the next filter through the pipes, creating a data flow pipeline.

1. **Filter:** All processes that contain a significant general algorithm that can be reused in other tasks. The filters can be re-ordered to solve different problems.
2. **Pipe:** Specific tasks that are required to communicate between the filters.

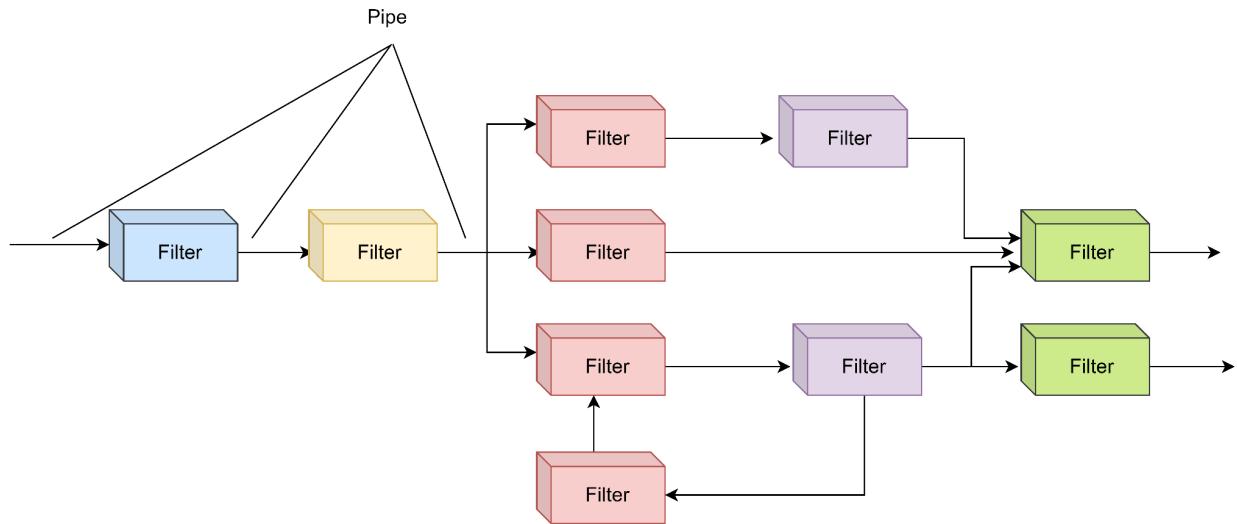


Fig B2 : Pipe And Filter Architecture

## 2.4 MVC Architecture

While widely used, MVC's current implementation is far from what it began with. We use the original MVC architecture recommended by Martin Fowler in his article. (**Reference:** Fowler, Martin. "GUI Architectures." *Martin Fowler*, 18 July 2006, <https://martinfowler.com/eaaDev/uiArchs.html>. Accessed 9 November 2023)

The model-view-controller architecture was developed with the motivation of separation of concern between the graphical user interface development and business logic development. It makes use of an observer pattern along with a controller in-between to facilitate one way communication between the graphical user interface and business logic to reduce coupling.

Let's define the three components of the MVC architecture.

1. **Model:** Contains business logic, persistence logic and every backend logic.
2. **View:** The graphical view that the user interacts with. It changes very often.

3. **Controller:** It captures user events and data and decides what to do with that data. It usually changes GUI against some user input/event. Or It may communicate with the model for some data/event.

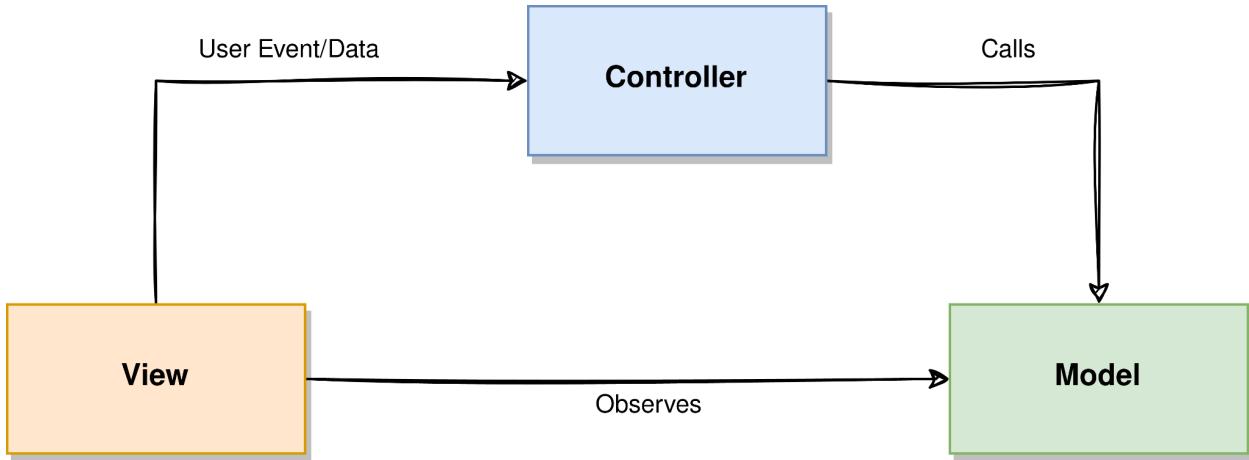


Fig B3: MVC Architecture

Model and View combined is called User Interface. User Interfaces subscribes to the model. When certain operations occur, the model notifies all the subscribers of that operation. This way, the views can update themselves whenever some changes occur in the model.

### 3. Top Level Components

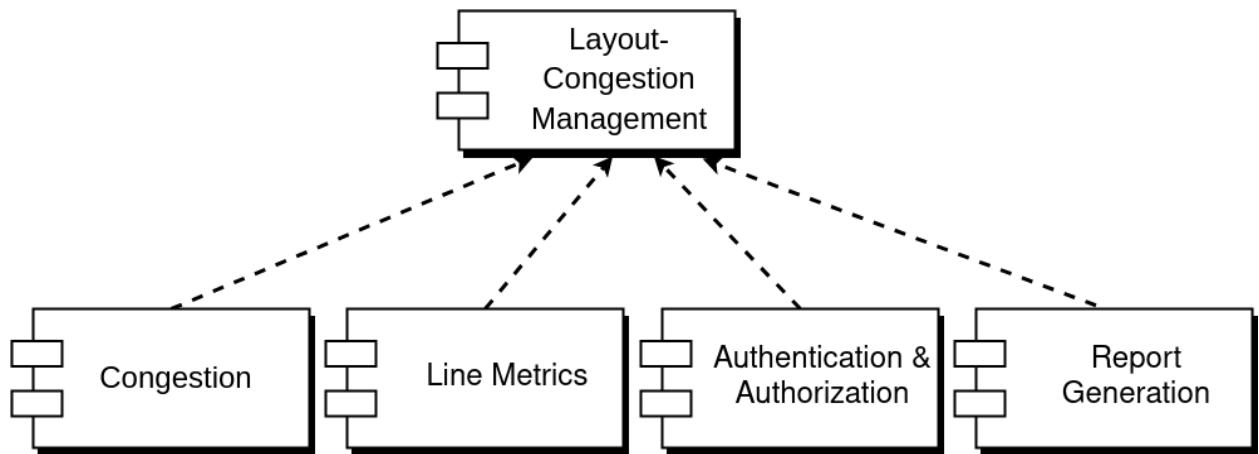


Fig- C1 : Top Level Components of LCM

#### Component Details

1. **Congestion:** The surveillance data is fetched from the camera, partitioned as necessary and then, congestion detection algorithms are used to identify the congestion for the respective workstation. When congestion is identified, the system flags the affected workstation and informs the line chief. The line chief then directs supervisors to analyze and document the causes of congestion, which could range from worker learning difficulties and machine issues to problems with line balancing.
2. **Line Metrics:** The system handles tasks related to lines, such as creating lines, parsing layout of lines, predicting the hourly production of a line etc.
3. **Authentication & Authorization:** The application allows the admin user to create and manage user accounts for different roles such as production manager, line chief, supervisor, and viewer. The admin can set usernames, designations, and general descriptions for each user and assign a default password. Employees of different roles can log in using those login credentials.
4. **Report Generation:** The system generates various reports, including hourly production reports, daily production reports, quality assurance reports, congestion reports, and line balancing reports. These reports provide insights into production performance, targets vs. achieved production, quality issues, congestion trends, and line balancing suggestions.

## 4. Instantiate Pattern

The components identified in Section-3 are broad. They need to be divided into smaller components to later convert them into sizable design classes. And just a collection of components doesn't create a meaningful design. So we apply architectural patterns on the components to make a maintainable collection of components.

### 4.1 Component : Congestion Detection

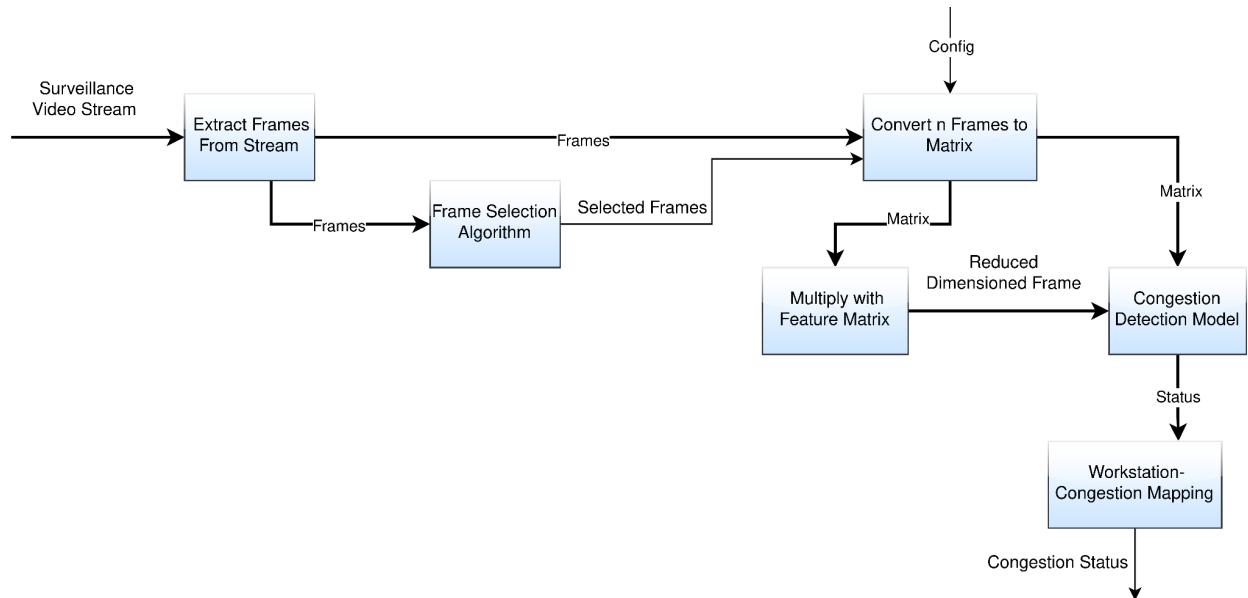


Fig D1: Congestion Detection using Pipe & Filter Pattern

Surveillance Video is streamed to the filter, *Extract Frames From Stream*. It extracts frames from the video stream at 30fps. Optionally, a *Frame Selection Algorithm* can be used to select fewer frames to boost efficiency. Then the frames are converted to vectors. Matrix multiplication may be performed to reduce the dimension of the image before sending it to the congestion detection model. The model predicts congestion status, which is then mapped to the workstations. The result is then sent through the output pipe.

## 4.2 Component : Report Generation

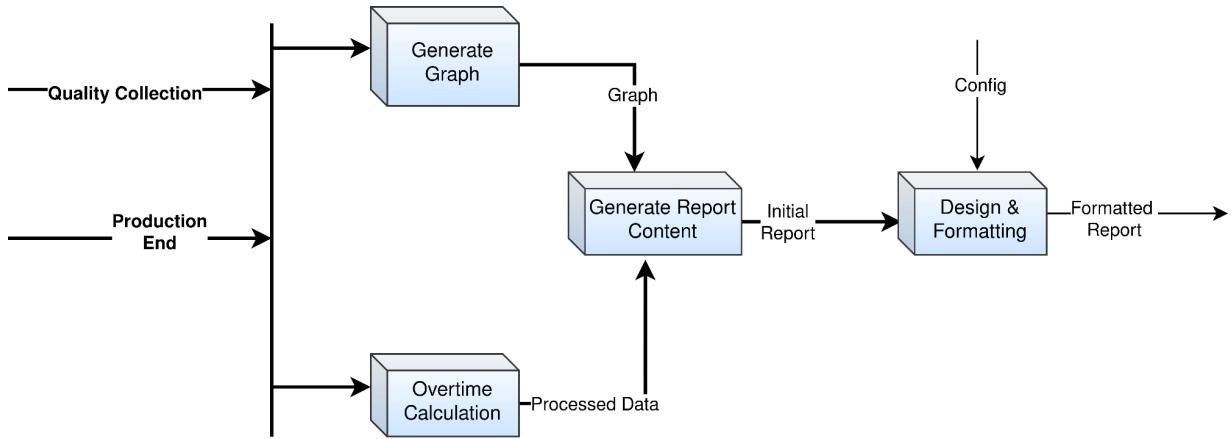


Fig D3: Pipe & Filter Architecture for Report Generation

The admin can generate reports for the overall project, which includes tasks such as data retrieval and formatting. The Pipe and Flow architecture allows these tasks to be broken down into smaller, manageable components (filters), each responsible for a specific data processing step. These filters can then be interconnected in a logical sequence (pipes) to create a cohesive pipeline for report generation.

## 4.3 Component : Congestion Issue

The Congestion Issue component is responsible for detecting, managing and addressing congestion-related problems within the assembly line. It uses MVC pattern. The users can

access different interfaces according to their roles. They use the controllers to

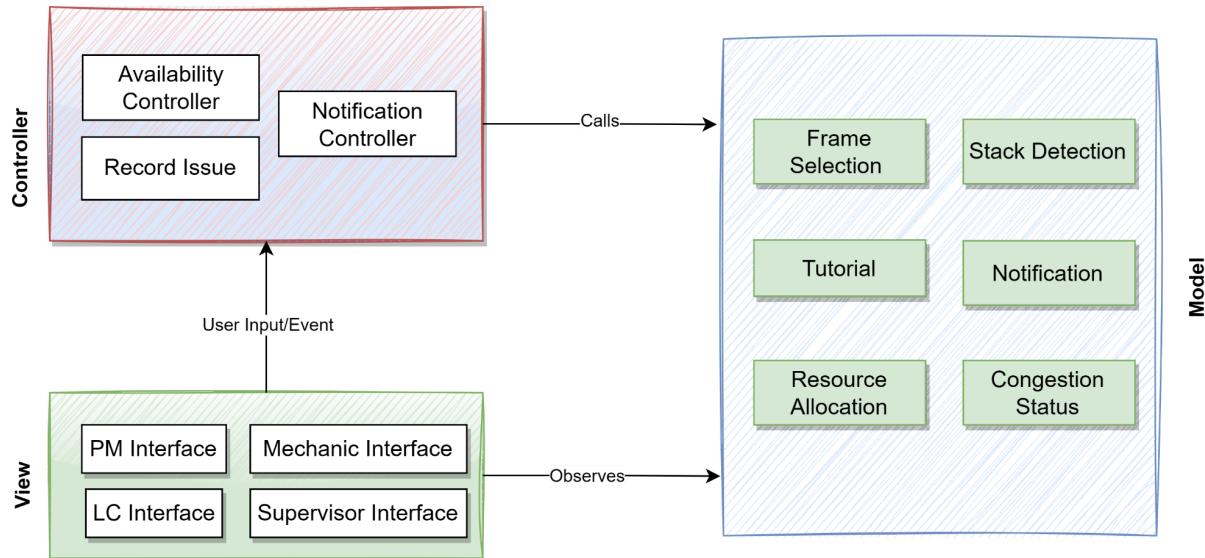


Fig D4: MVC Architecture for Congestion Issue

## 4.4 Component : Line Metrics

Line Metrics include: Efficiency, Predicted Production = Number of products that can be produced with current layout \*(1-error percentage)

The Pipe and Flow architectural design is particularly suitable for Line Metrics due to its efficiency, modularity, and scalability. This pattern leverages the strengths of a streamlined and sequential data processing flow, making it an ideal choice for managing and optimizing assembly line metrics.

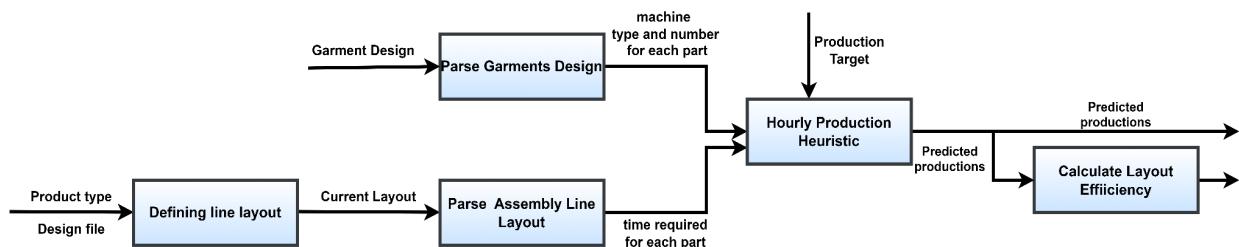


Fig D5: Pipe & Filter Architecture for Line Metrics

## 4.5 Component : Authorization & Authentication

Layered architectural design is well-suited for implementing Authorization and Authentication in an assembly line management system due to its modular structure, security isolation, scalability, reusability, flexibility, and ease of maintenance. This approach ensures that the upper layers can work without knowing explicit implementation details of the lower layers, which makes managing the authentication and authorization process easier as data passes only from lower layer to upper layer.

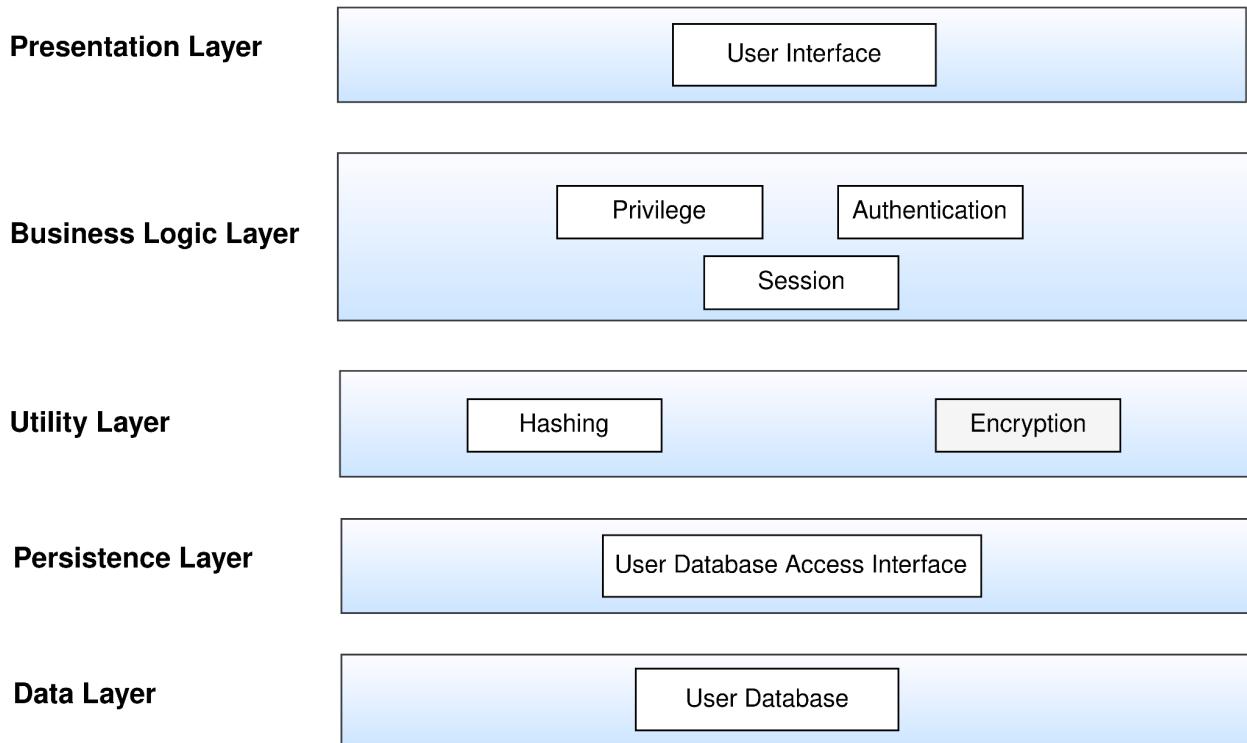


Fig D6: Layered Architecture for Authentication & Authorization

## 5. From Analysis Classes to Design Classes

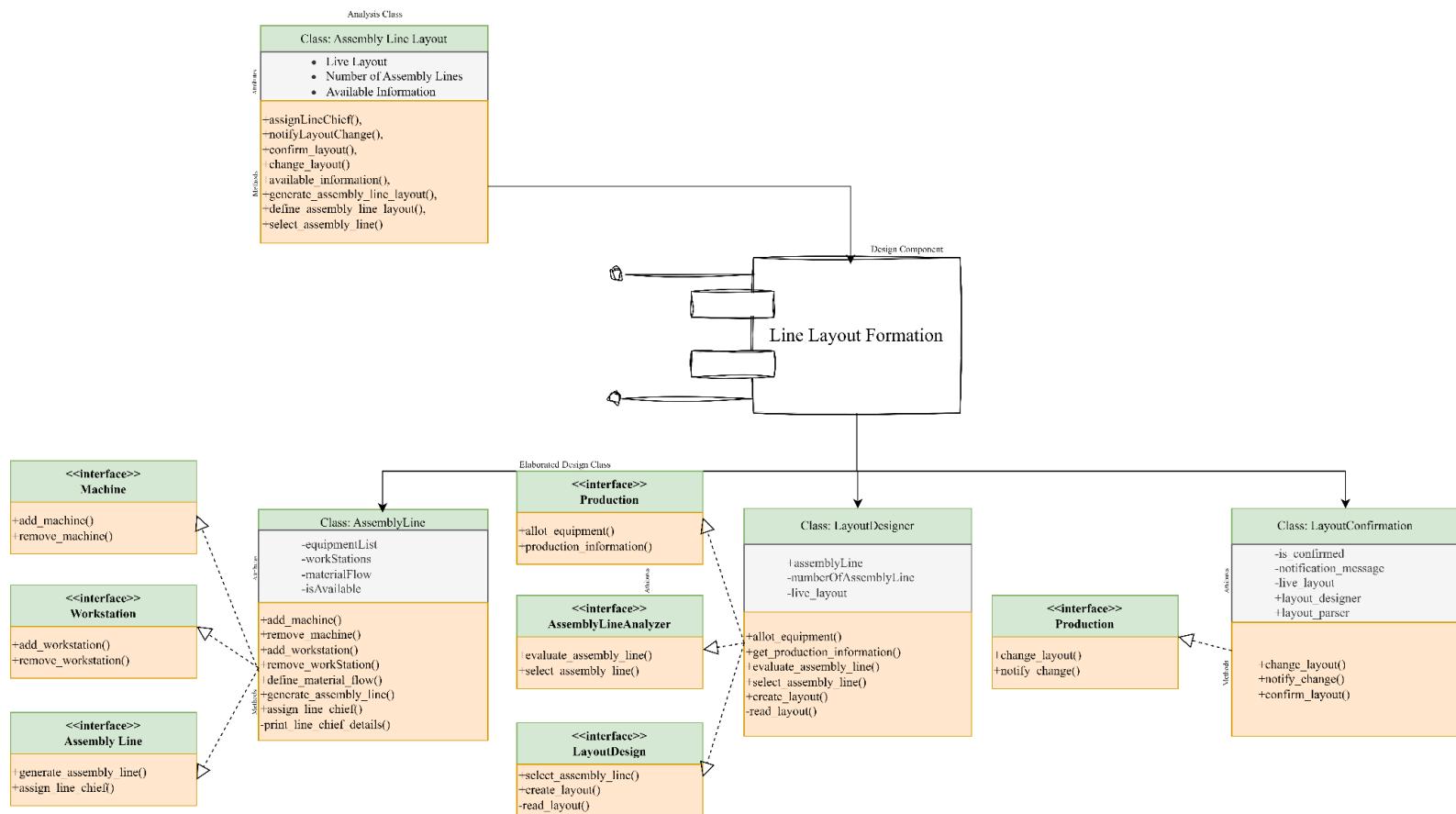
From the SRS of the LCM we get the analysis classes. We now need to associate these classes to components:

1. Authentication and Authorization (Component): User (Class)
2. Congestion: Congestion + Issue
3. Report Generation: Report
4. Line Layout (Layout + Metrics): Line Balancing + Assembly Line Layout

| Old Component                           | Merged into  | Associated Analysis Class        |
|---|--------------|----------------------------------|
| Authentication and Authorization        | N/A          | User                             |
| Congestion Issue + Congestion Detection | Congestion   | Congestion, Issue, Resource      |
| Report Generation                       | N/A          | Report                           |
| Layout Assembly + Line Metrics          | Line Metrics | LineBalancing, AsseblyLineLayout |

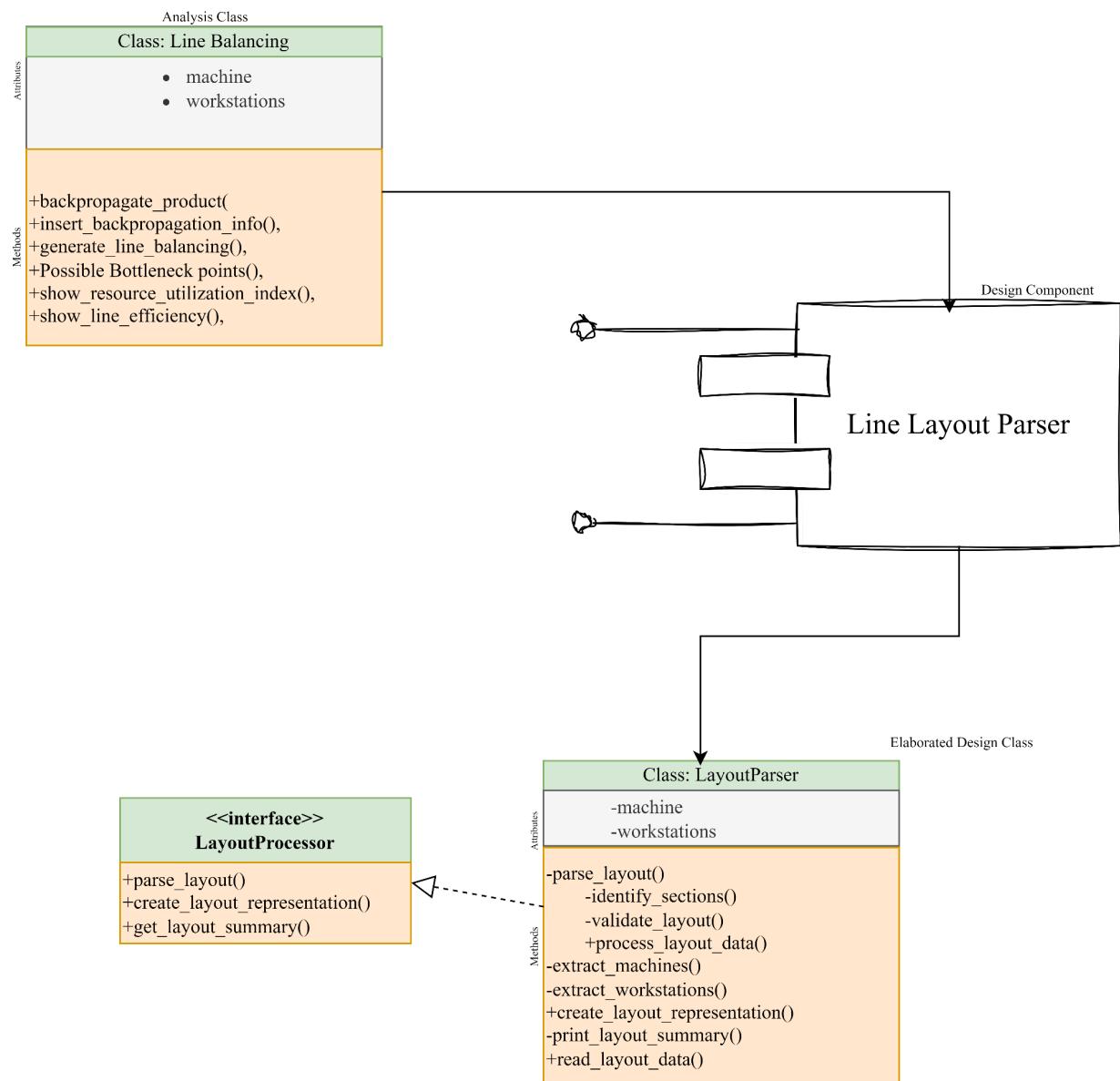
## 5.1 Design Class for Line Metrics Component

### 5.1.1 Line Layout Formation

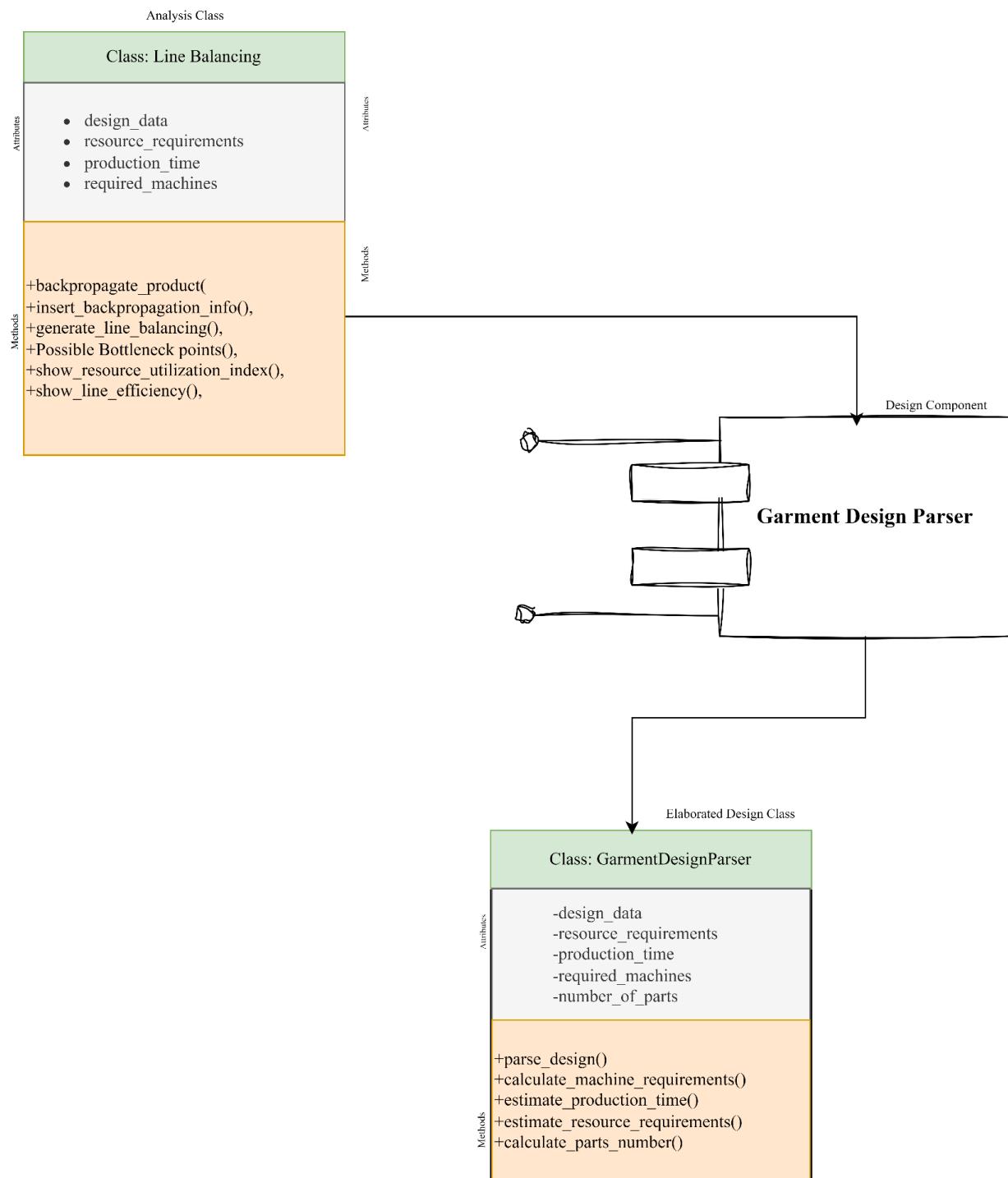


For better quality image, please visit this link: [Layout and line balancing drawio](#)

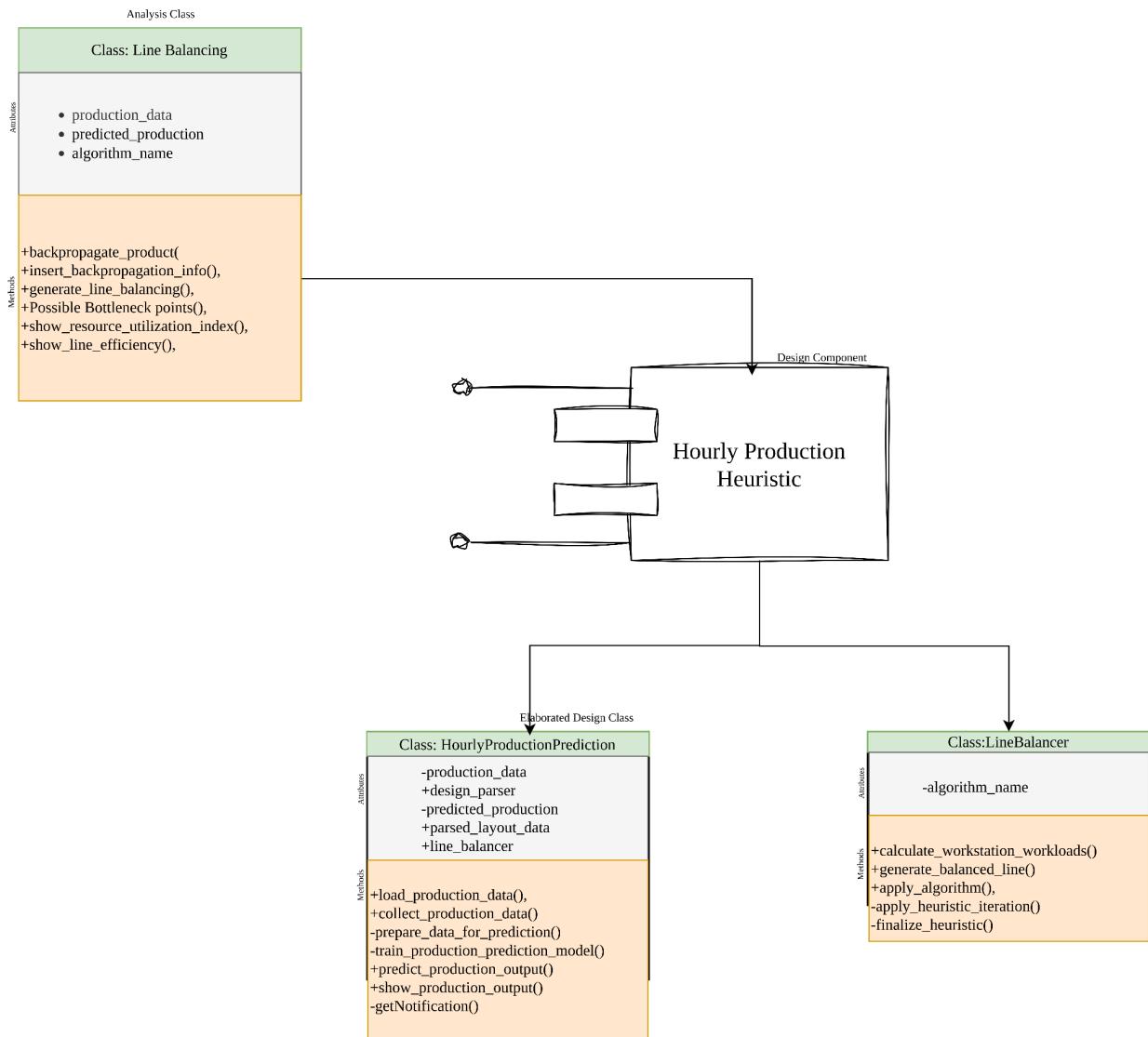
## 5.1.2 Line Layout Parser



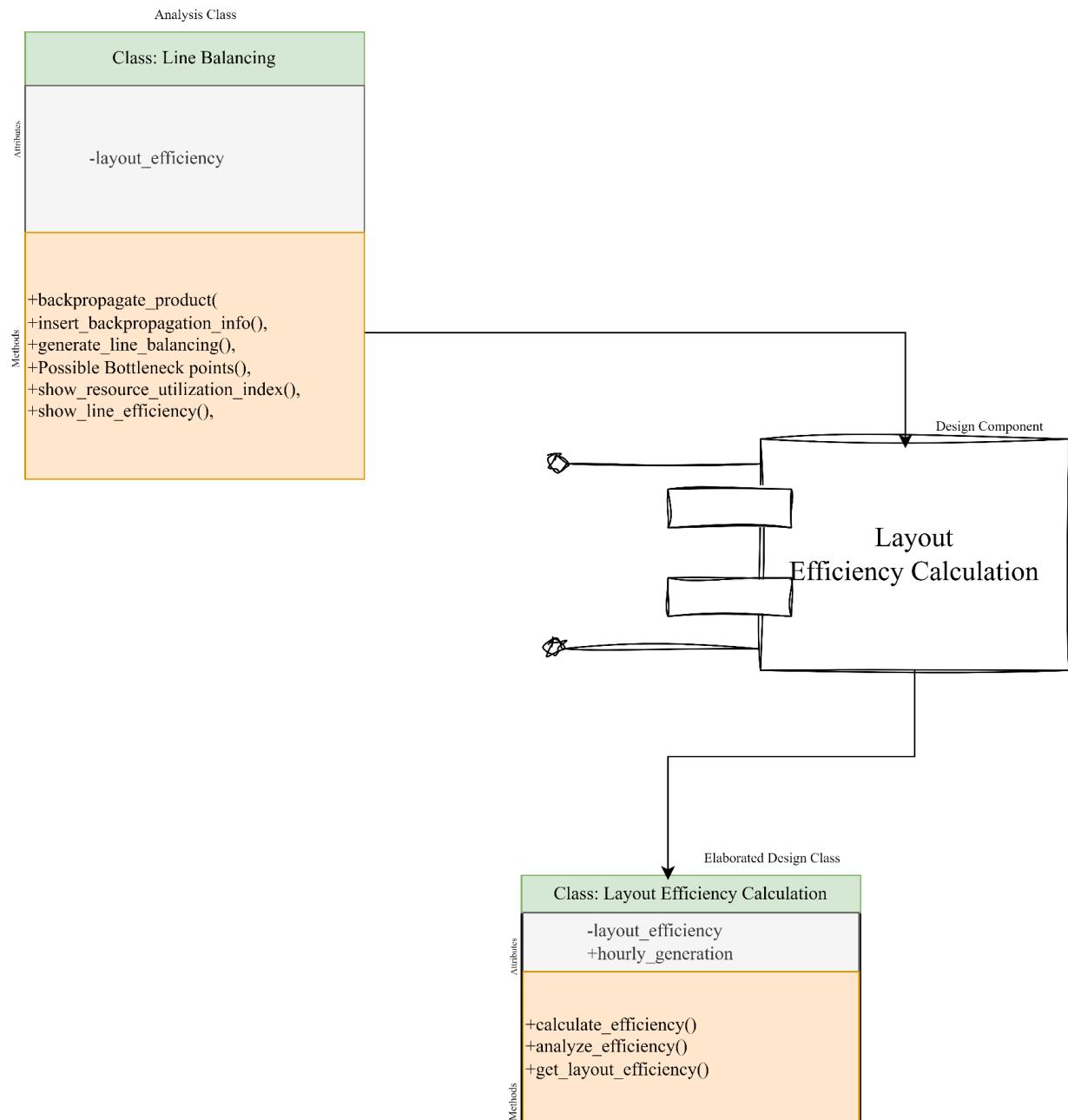
### 5.1.3 Garment Design Parser



## 5.1.4 Hourly Production Heuristic

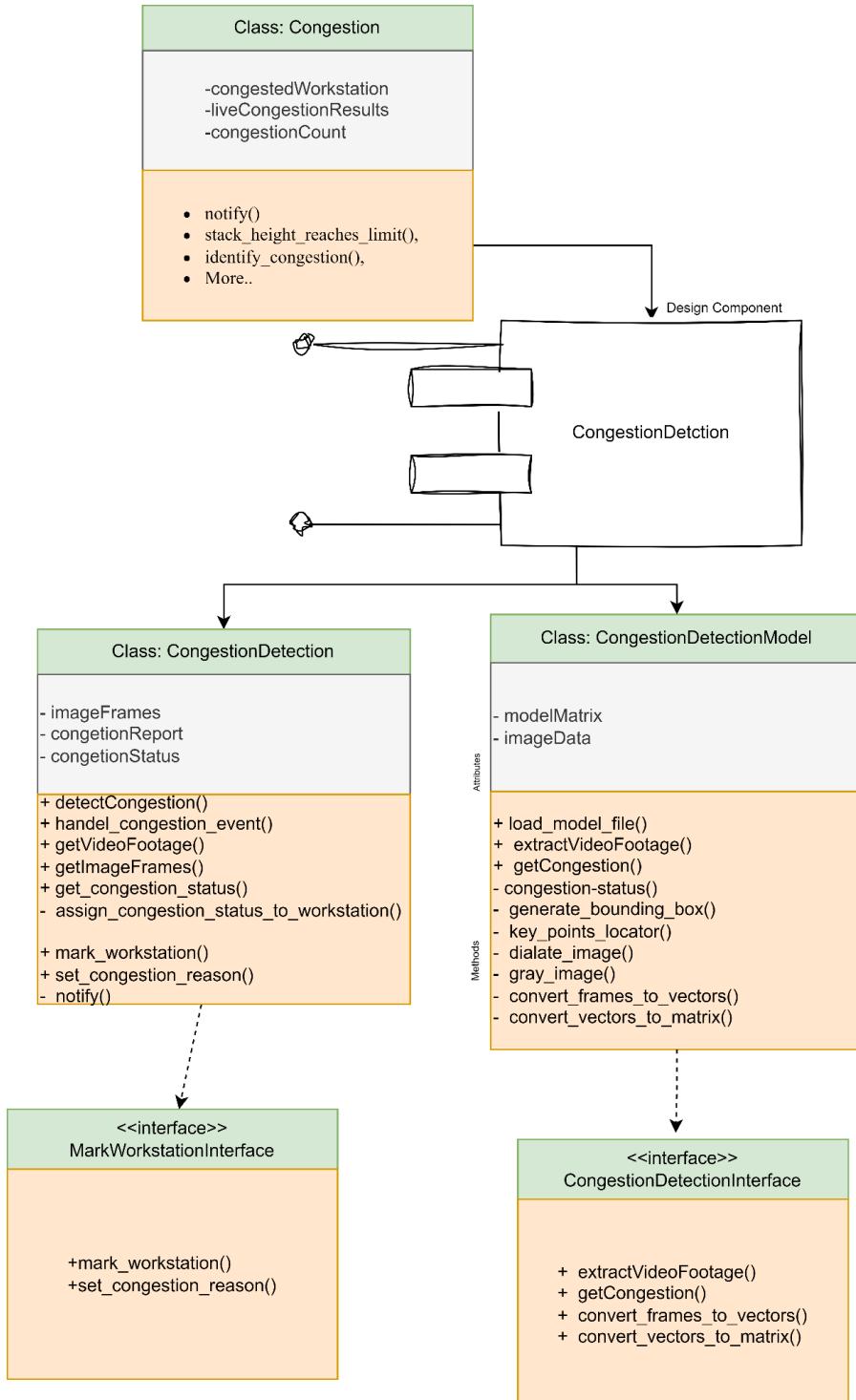


## 5.1.5 Layout Efficiency Calculation

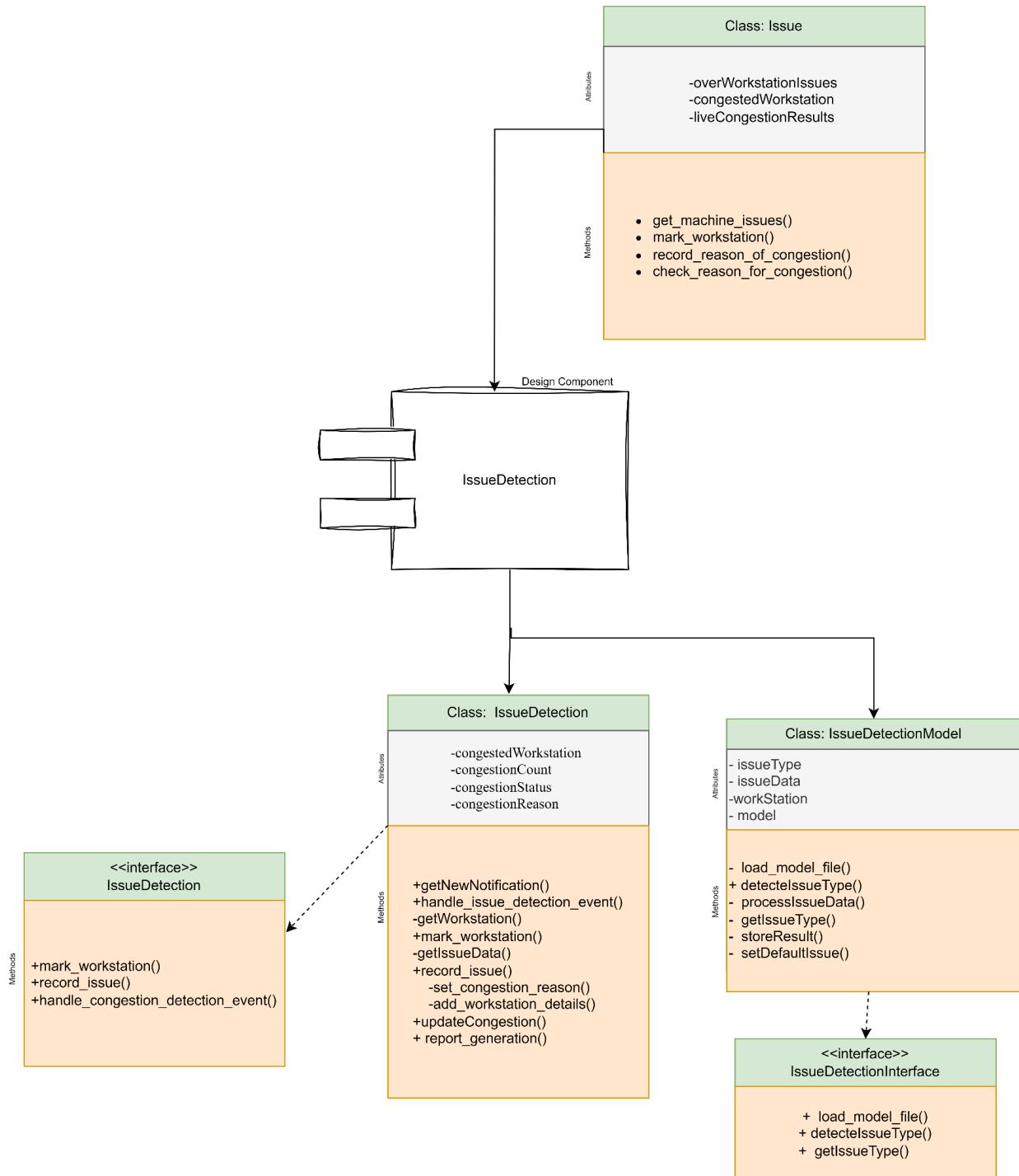


## 5.2 Design Class for Congestion

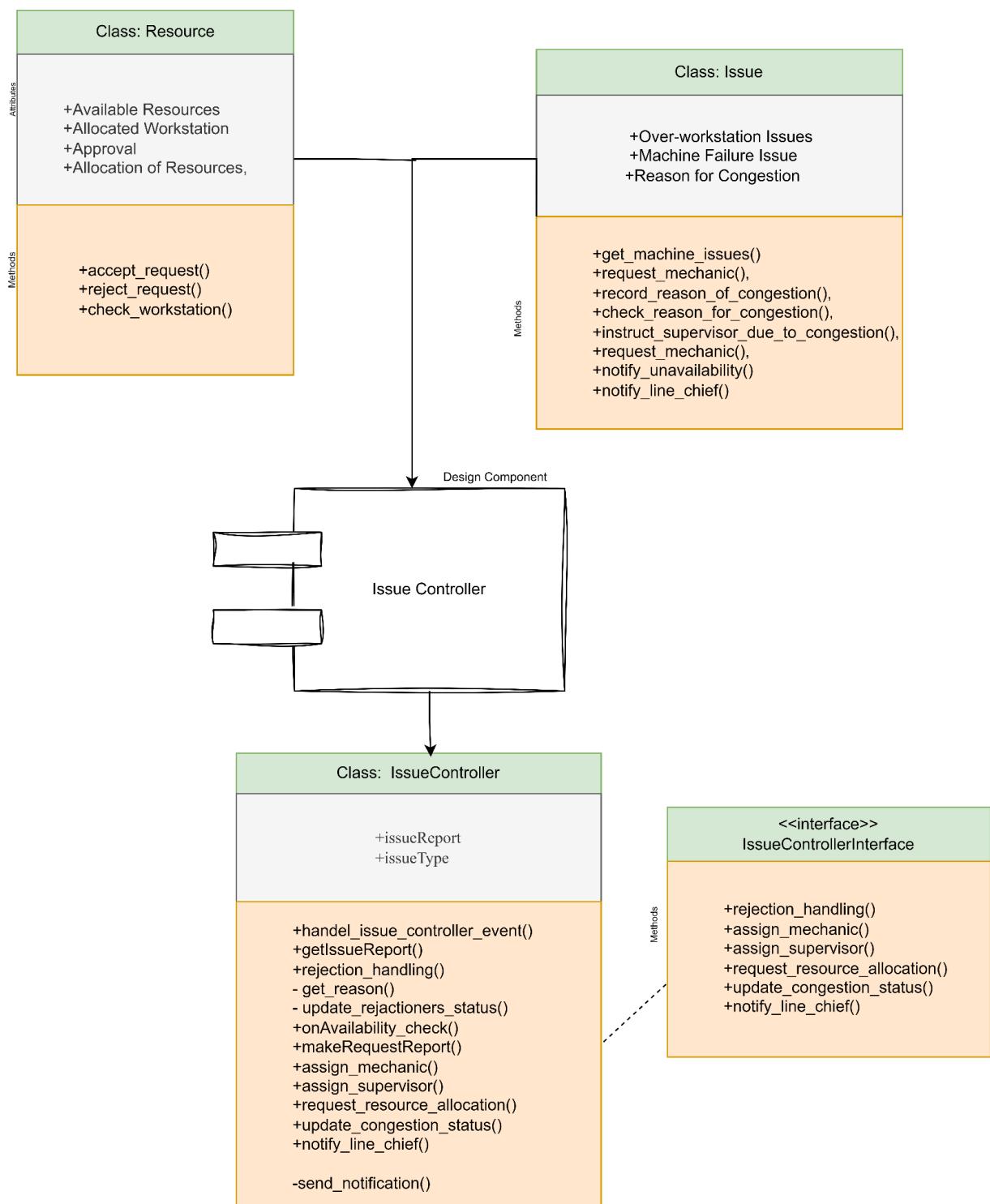
### 5.2.1 Congestion Detection



## 5.2.2 Issue Detection



### 5.2.3 Issue Controller



## 5.3 Design Class For User

### 5.3.1 Authentication

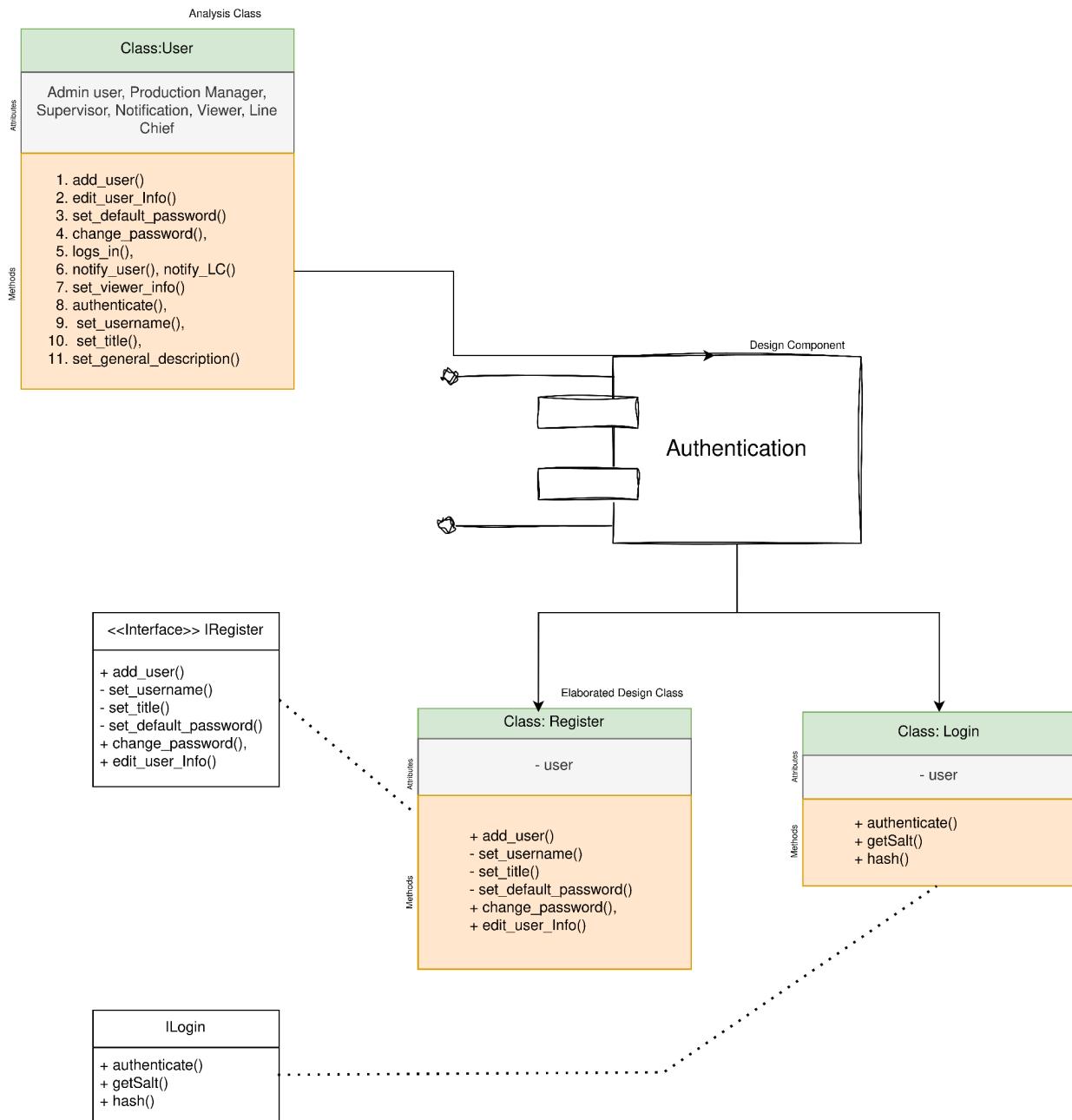


Fig: Elaborating Design Classes For Authentication Component

Note that, we have renamed the “logs\_in()” method to “authenticate()”. Additionally, we have elaborated some analysis class methods into template methods.

### 5.3.2 Privilege

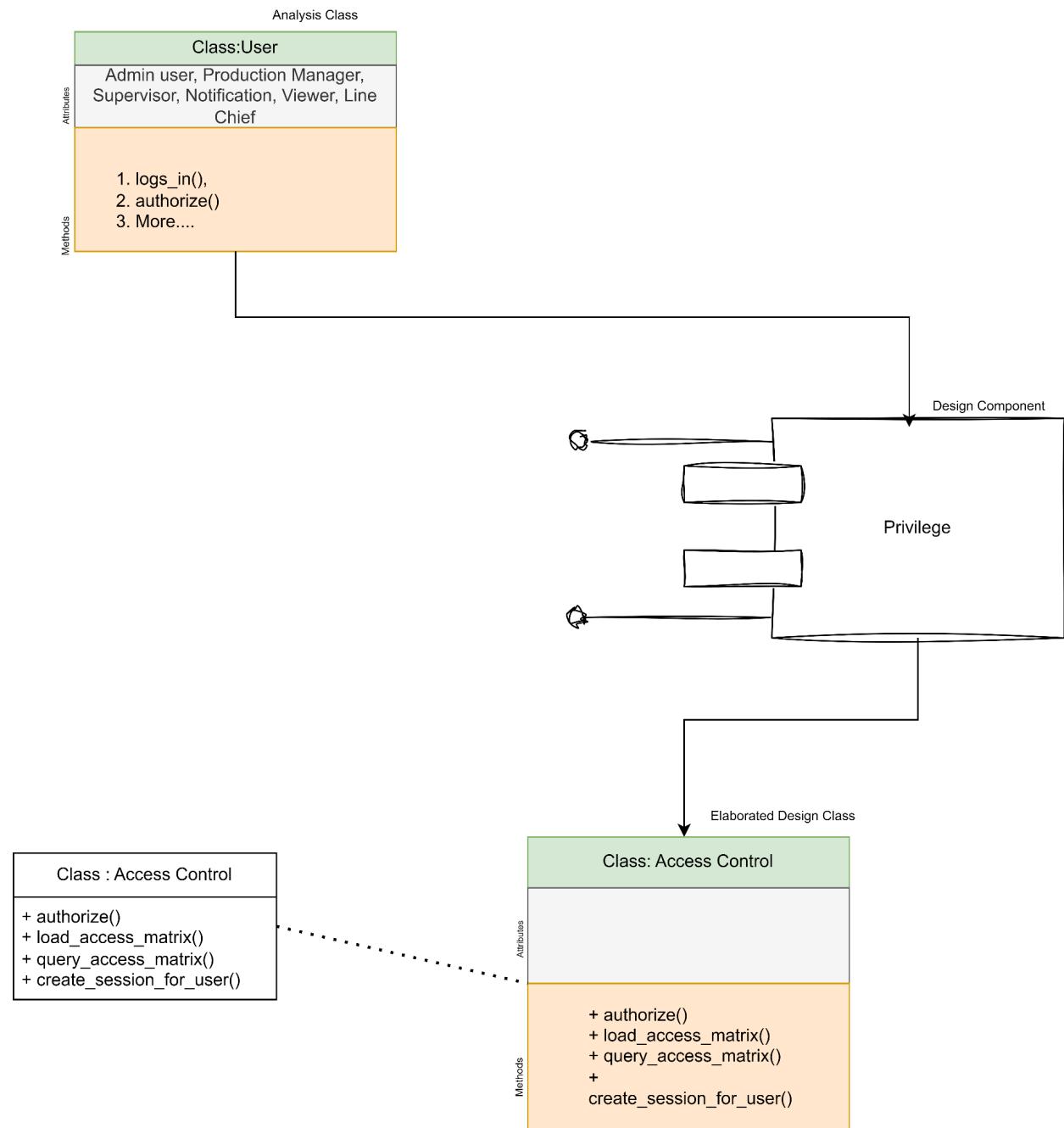


Fig: Elaborating Design Classes For Privilege Component

Here, we expanded the `logs_in()` method into this class due to implementation necessity.

### 5.3.3 Hashing

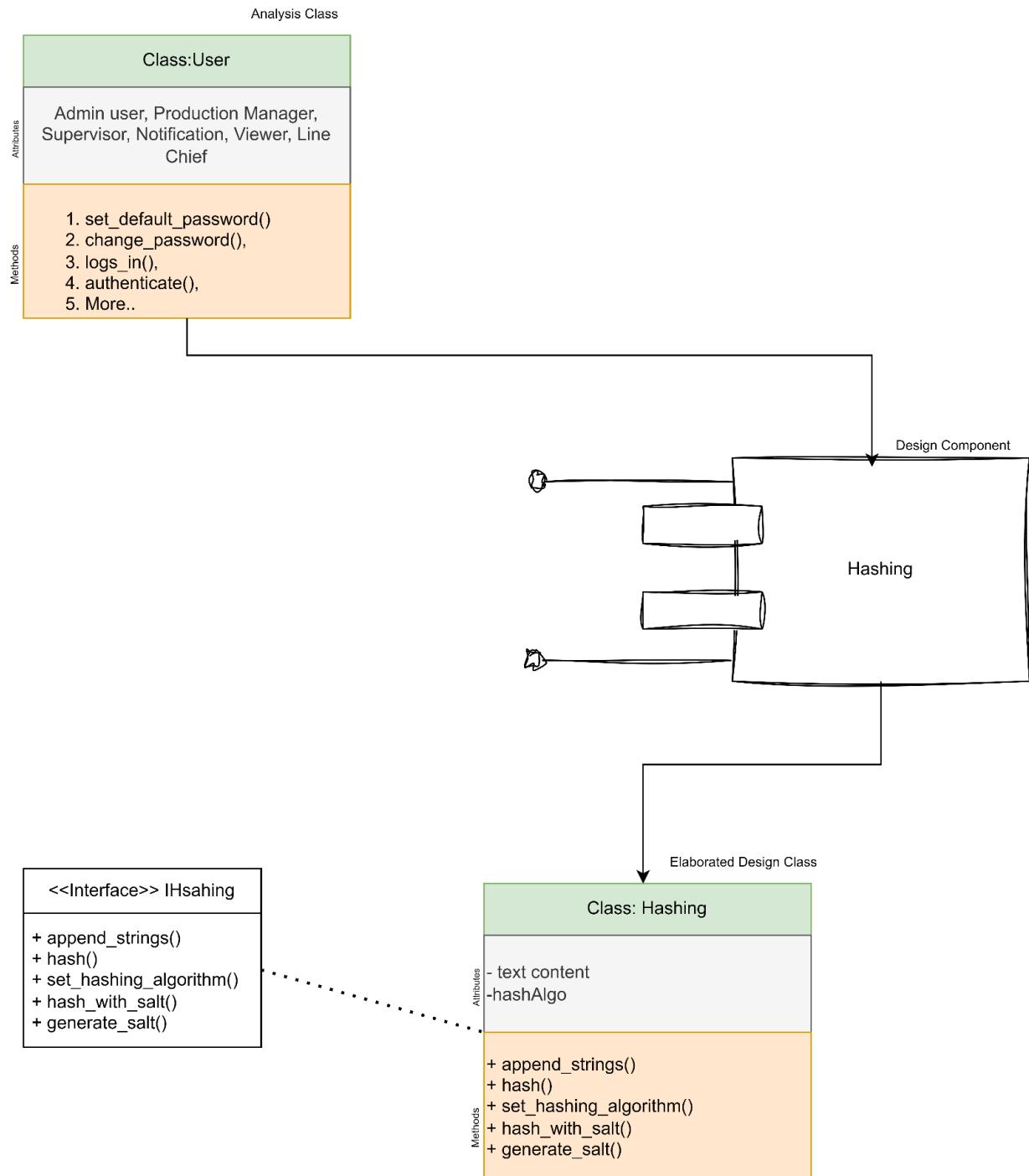


Fig: Elaborating Design Classes For Hashing Component

As we did previously, we expanded the analysis class methods for implementation necessity.

### 5.3.4 Encryption

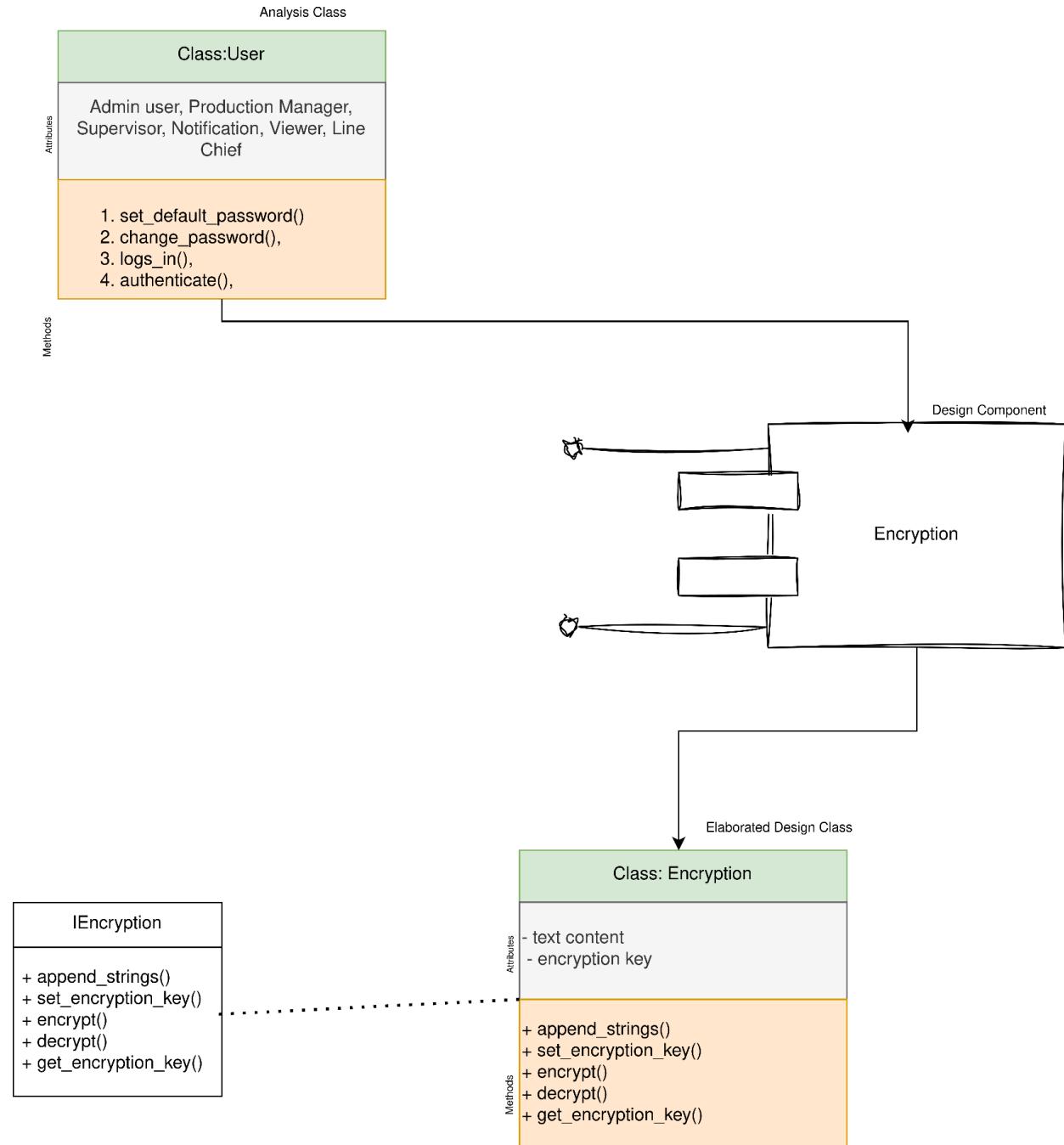
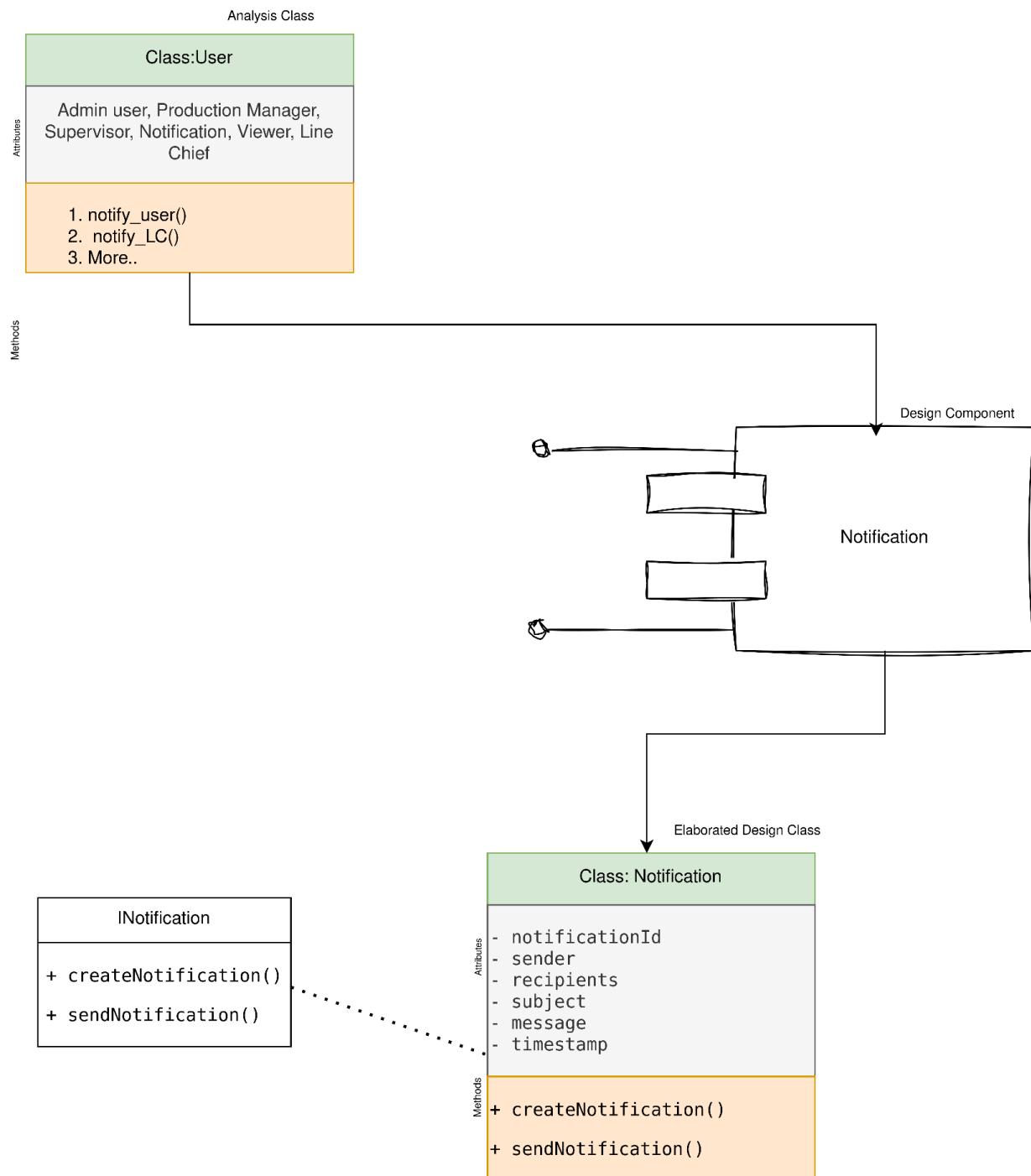


Fig: Elaborating Design Classes For Encryption Component

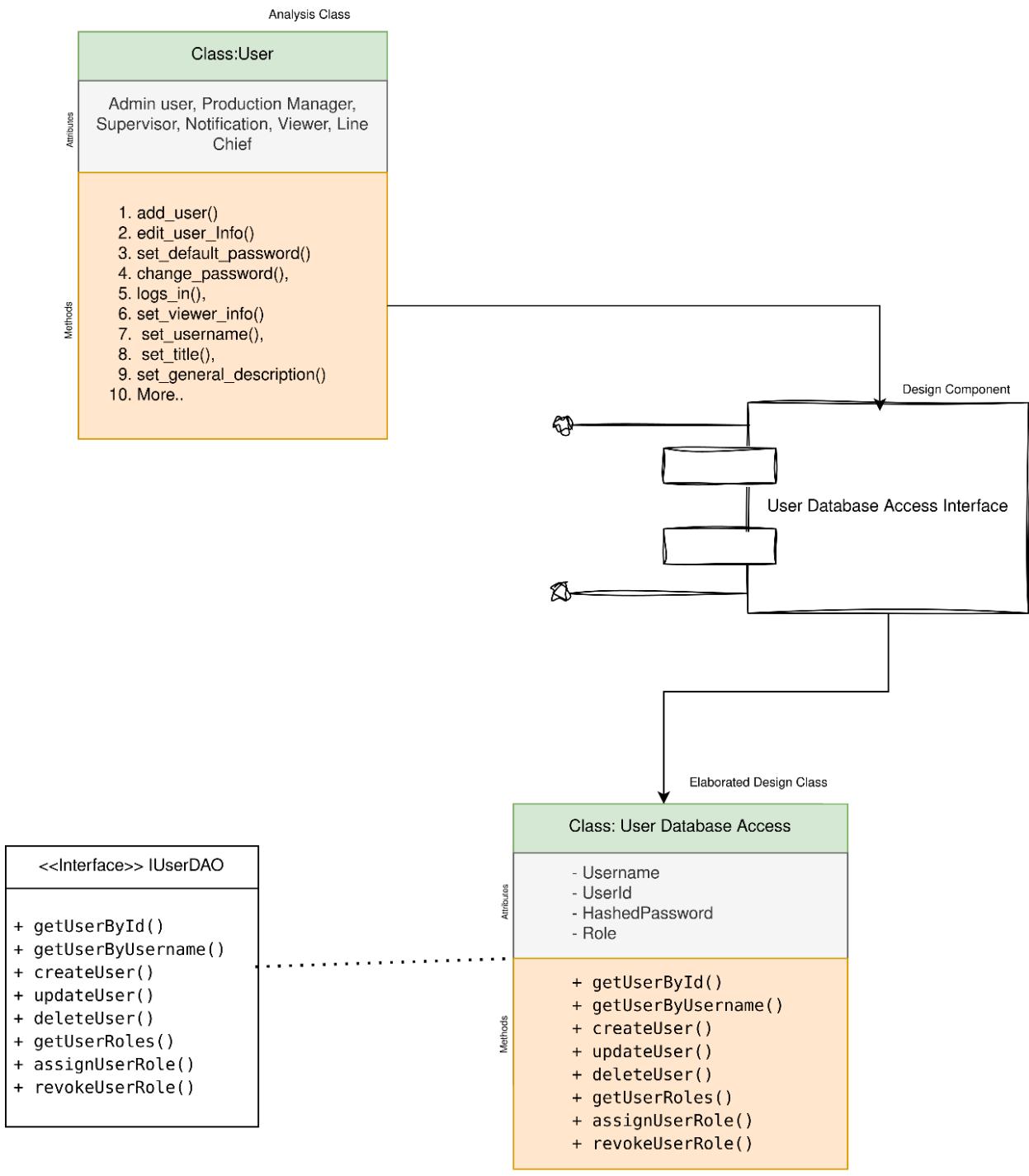
As we did previously, we expanded the analysis class methods for implementation necessity.

### 5.3.5 Notification



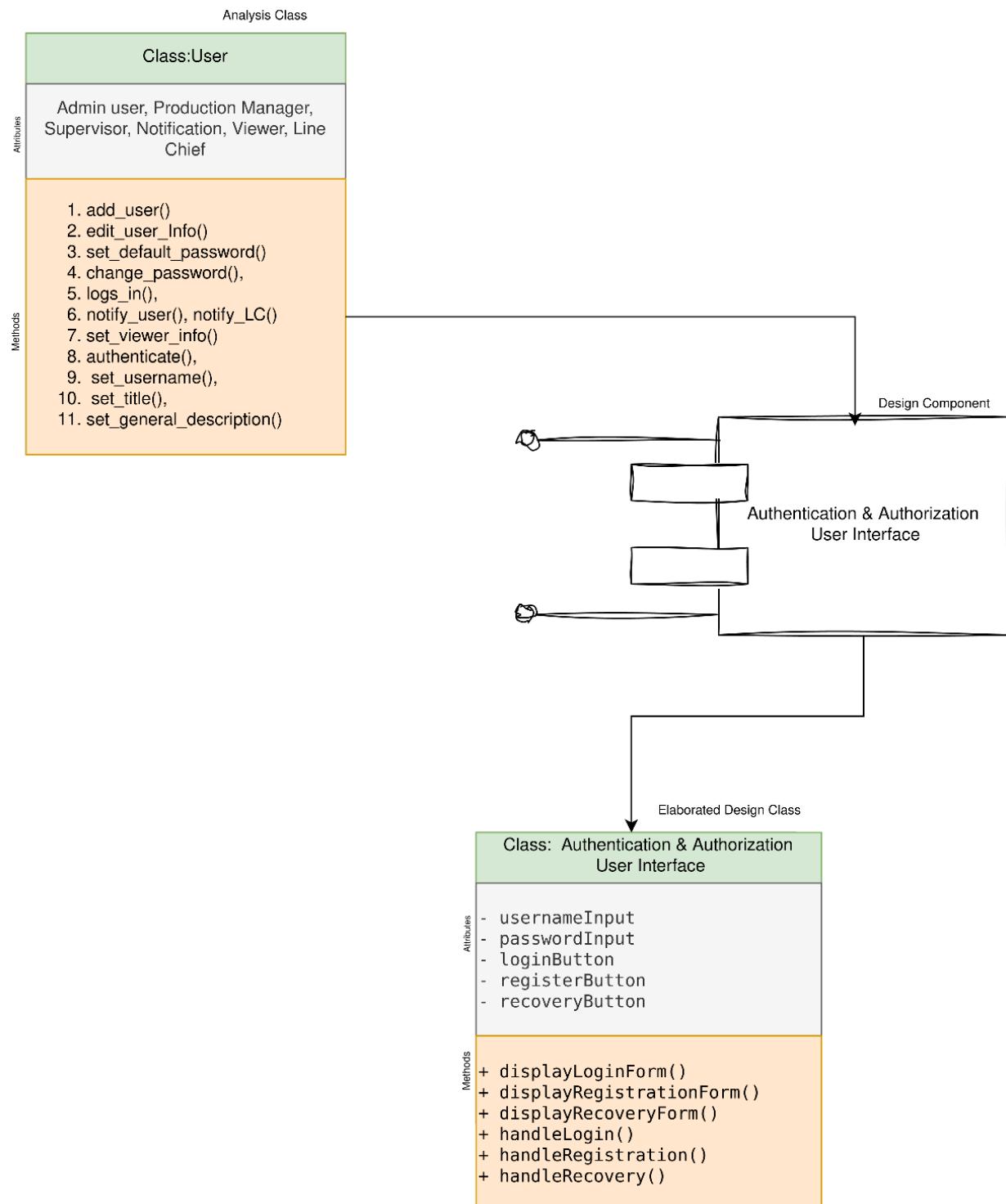
Here, we renamed the methods from the analysis class for better understanding of the class.

### 5.3.6 User Database Access Interface



Here, almost all methods in the User Analysis class uses User DAO class somehow and hence, we show them in the analysis class diagram above.

### 5.3.7 - Authentication & Authorization User Interface (UI)

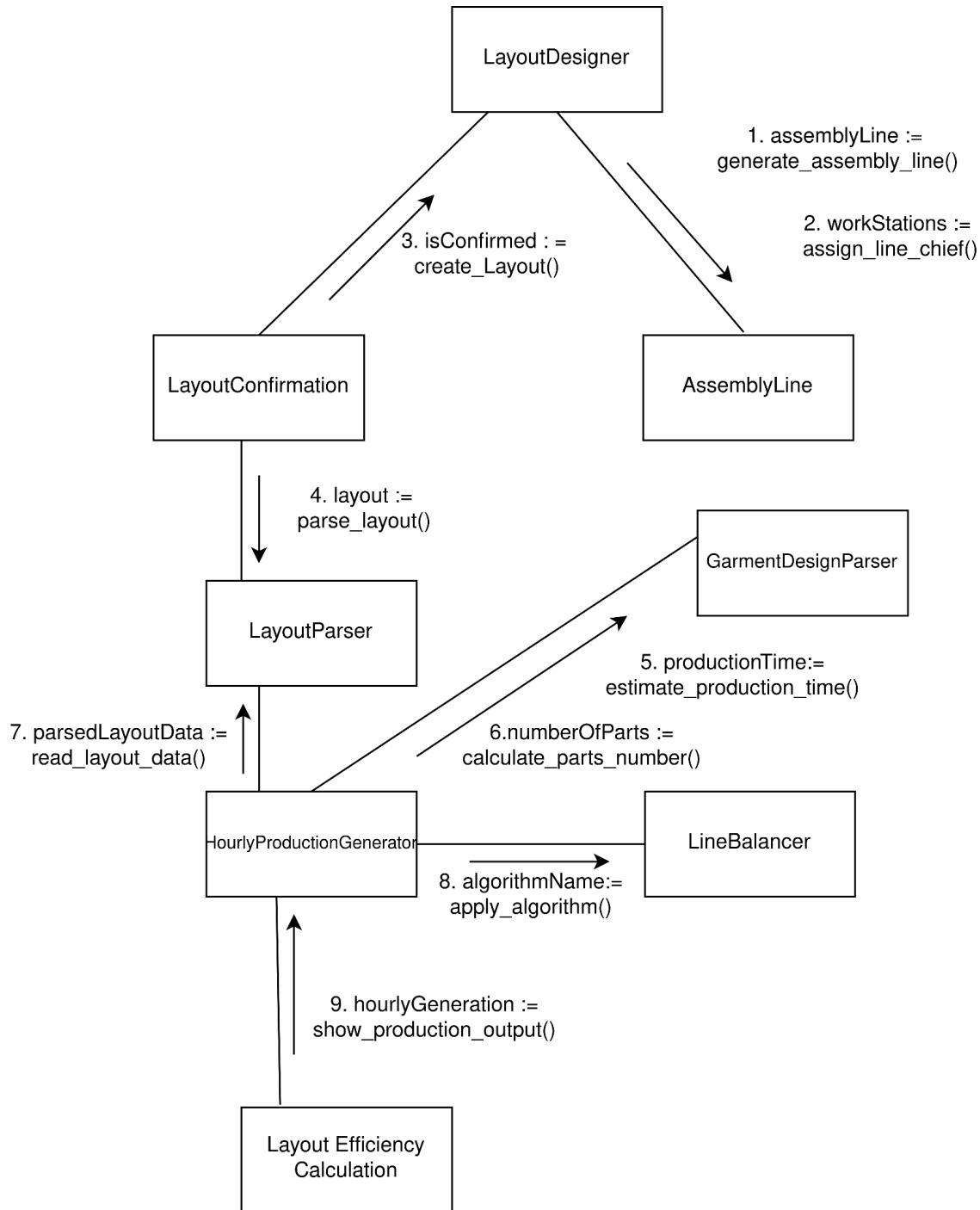


We did not implement any interface here because we believe that GUI classes are susceptible to changes and are not called by any non-library modules. Hence, an interface is unnecessary.

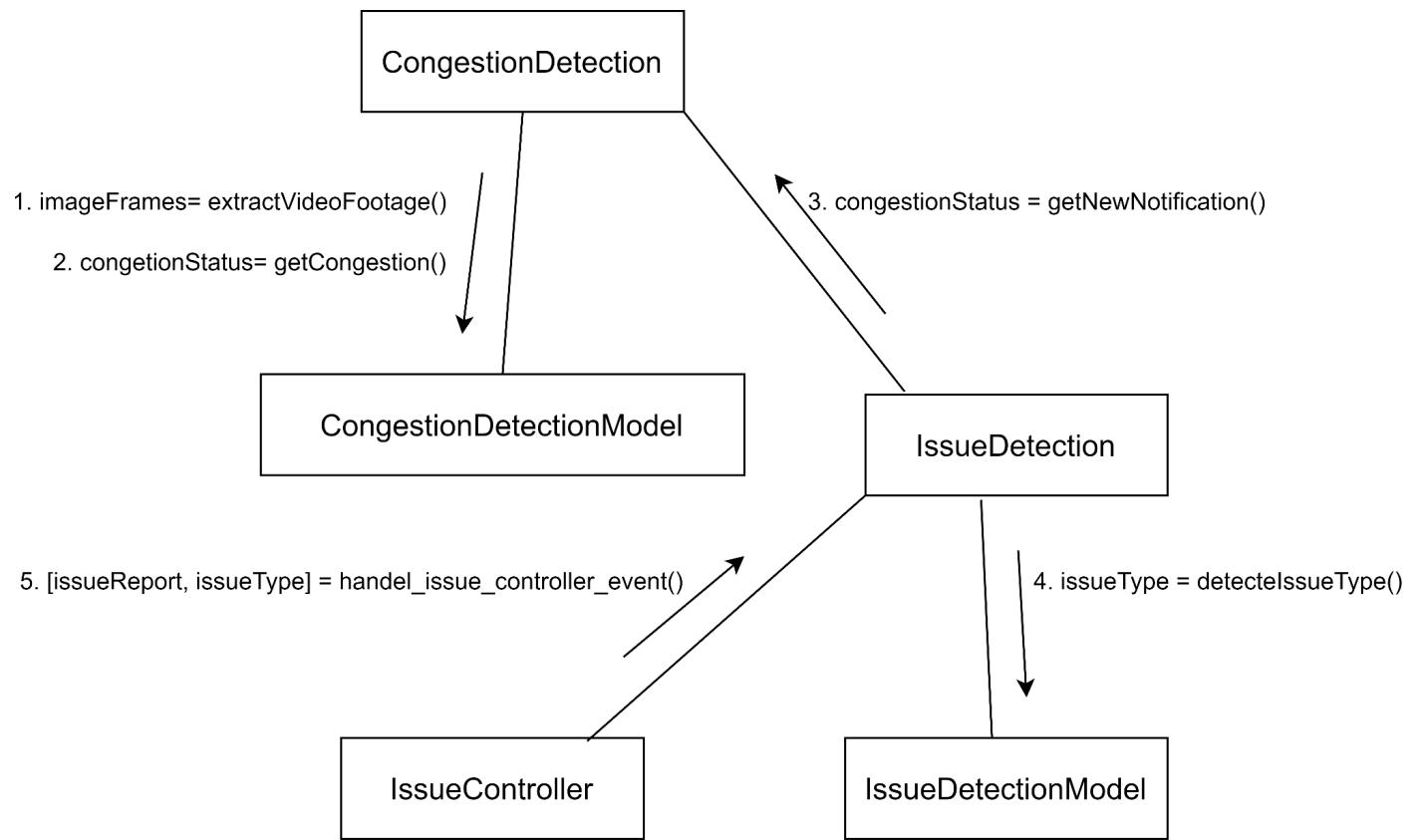
# 6. Message Passing between Classes

Each of the boxes represents a class. The arrows indicate that, a method is being called from the class at the arrow's end and the method originates from the class at the tip of the arrow.

## 6.1 Line Metrics Component



## 6.2 Congestion Component



## 7. Elaborate attributes and define data types

### 7.1 Line Metrics Component

| Class              | Elaborated Attributes   |
|--------------------|---|
| AssemblyLine       | equipmentList: list of strings = [] {stores names of different equipment available in the assembly line}<br><br>workStations: list of strings = [] {stores names of different workstations in the assembly line}<br><br>materialFlow: string = "" {describes the flow of materials in the assembly line}<br><br>isAvailable: boolean = true {indicates whether the assembly line is currently available or not}   |
| LayoutDesigner     | assemblyLine: AssemblyLine = new AssemblyLine() {references the assembly line associated with the layout designer}<br><br>numberOfAssemblyLine: integer = 0 {represents the count of assembly lines the designer is working on}<br><br>live_layout: string = "" {stores the live layout data being designed}  |
| LayoutConfirmation | is_confirmed: boolean = false {indicates whether the layout is confirmed or not}<br><br>notification_message: string = "" {contains any notification messages related to layout confirmation}<br><br>live_layout: string = "" {stores the live layout data}<br><br>layout_designer: LayoutDesigner = new LayoutDesigner() {references the layout designer associated with the layout confirmation}<br><br>layout_parser: LayoutParser = new LayoutParser() {references the layout parser associated with the layout confirmation} |
| LayoutParser       | machine: string = "" {stores information about the machines parsed from the layout}   |

|                               |   |
|-------------------------------|---|
|                               | workstations: list of strings = [] {stores information about the workstations parsed from the layout}   |
| GarmentDesignParser           | <p>design_data: string = "" {stores data related to garment design}</p> <p>resource_requirements: list of strings = [] {stores information about the resource requirements for garment production}</p> <p>production_time: string = "" {stores the time required for garment production}</p> <p>required_machines: list of strings = [] {stores names of machines required for garment production}</p> <p>number_of_parts: integer = 0 {represents the count of parts in the garment design}</p>  |
| HourlyProductionGenerator     | <p>production_data: string = "" {stores data related to hourly production}</p> <p>design_parser: GarmentDesignParser = new GarmentDesignParser() {references the garment design parser associated with the hourly production generator}</p> <p>predicted_production: string = "" {stores the predicted production data}</p> <p>layout_parser: LayoutParser = new LayoutParser() {references the layout parser associated with the hourly production generator}</p> <p>line_balancer: LineBalancer = new LineBalancer() {references the line balancer associated with the hourly production generator}</p> |
| LineBalancer                  | <p>algorithm_name: string = "" {stores the name of the algorithm used for balancing the line}</p>   |
| Layout Efficiency Calculation | <p>Layout Efficiency Calculation: float = 0.0 {stores the efficiency of the layout calculation}</p> <p>hourly_generation: string = "" {stores the hourly generation data}</p>   |

## 7.2 Congestion Component

| <b>Class</b>               | <b>Elaborated Attributes</b>   |
|----------------------------|--|
| IssueDetectionController   | <p>WorkstationList: list of strings = [] {stores names of different workstations available}</p> <p>CongestedWorkstation: list of strings = [] {stores names of different workstations that are congested}</p> <p>CongestionStatus: boolean = false {indicates if congestion results are live}</p> <p>CongestionCount: int = 0 {number of congestions detected}</p> |
| CongestionController       | <p>CongestedWorkstation: list of strings = [] {stores names of different workstations that are congested}</p> <p>CongestionStatus: boolean = false {indicates if congestion results are live}</p> <p>CongestionCount: int = 0 {number of congestions detected}</p>   |
| CongestionReasonController | <p>CongestionStatus: boolean = false {indicates if congestion results are live}</p> <p>CongestionReason: string = "" {congestion reason}</p>   |
| AvailabilityChecker        | <p>MechanicAvailability: boolean = true {indicates mechanic availability}</p> <p>WorkstationAvailability: boolean = true {indicates workstation availability}</p>  |
| Notification               | <p>CongestionIssue: string = "" {description of the congestion issue}</p> <p>Recipients: List&lt;string&gt; = [] {list of recipients for the notification}</p>   |
| AssemblyLineLayoutGUI      | <p>ProductionTarget: int = 0 {target production for the assembly line}</p> <p>CurrentProduction: int = 0 {current production on the assembly line}</p>   |

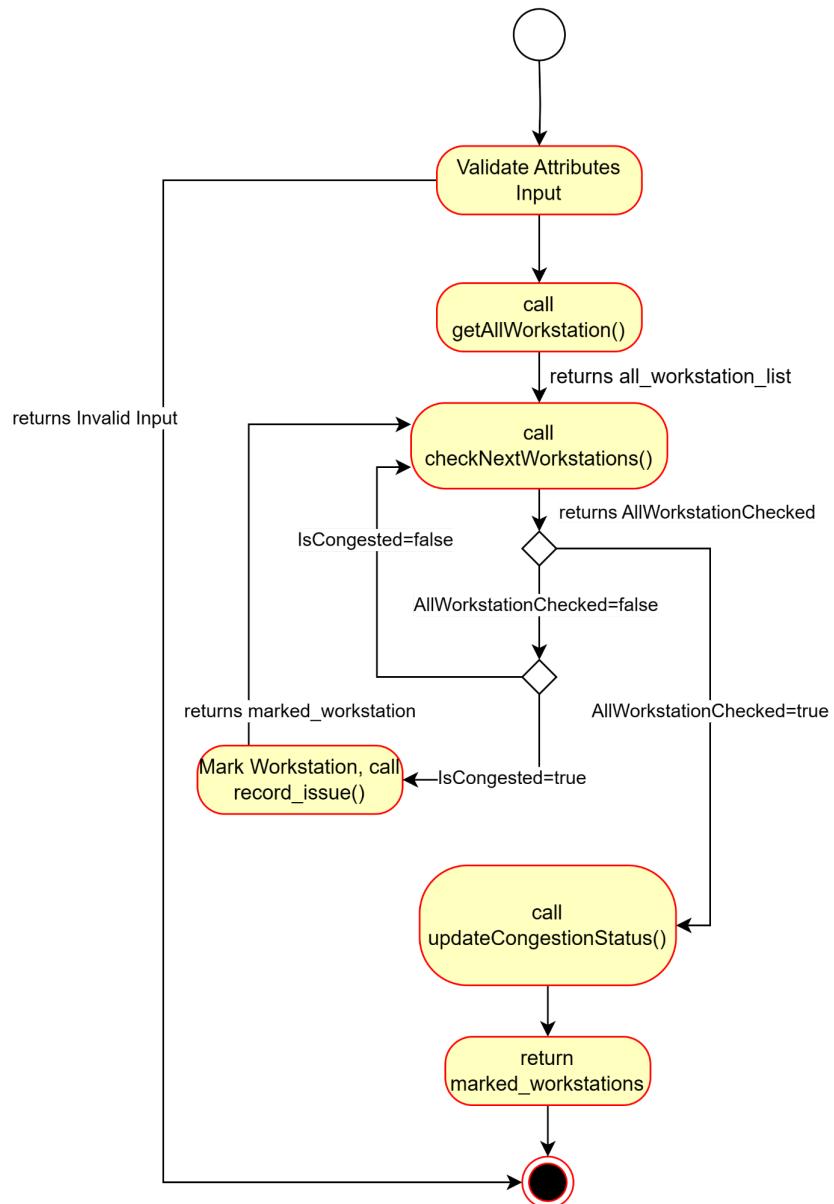
|                              |  |
|------------------------------|--|
|                              | <p>Resource:</p> <p>Name: string = "" {name of the resource}</p> <p>Type: string = "" {type of the resource}</p> <p>HourlyProduction: int = 0 {hourly production of the resource}</p>                          |
| ReportGUI                    | <p>MachineDetails: string = "" {details of the machines}</p> <p>HourlyProduction: int = 0 {hourly production}</p>  |
| ProductionManagerGUI         | <p>AmountOfResources: int = 0 {amount of resources available}`</p> <p>AssemblyLineLayout: AssemblyLineLayoutGUI {instance of the Assembly Line Layout GUI}`</p>  |
| Supervisor Congestion GUI    | <p>CongestedWorkstation: list of strings = [] {stores names of different workstations that are congested}</p> <p>LiveCongestionResults: boolean = false {indicates if congestion results are live}</p>         |
| Supervisor Submit Report GUI | <p>MachineDetails: string = "" {details of the machine}</p> <p>HourlyProduction: int = 0 {hourly production}</p> <p>CongestionStatus: boolean = false {indicates if congestion results are live}</p>           |
| Supervisor GUI               | <p>CongestionStatus: boolean = false {indicates if congestion results are live}</p> <p>ReportDetails: string = "" {details of the report}</p>  |
| Mechanic GUI                 | <p>MechanicAvailability: boolean = true {indicates mechanic availability}</p> <p>MachineStatus: string = "" {status of the machine}</p> <p>ReportMachineIssue: function {method to report a machine issue}</p> |

|                             |   |
|-----------------------------|---|
| Line Chief Congestion GUI   | <p>CongestedWorkstation: string = "" {name of the congested workstation}`</p> <p>LiveCongestionResults: boolean = false {indicates if congestion results are live}</p> <p>CongestionCount: int = 0 {number of congestions detected}</p>   |
| LC Assembly Line Layout GUI | <p>ProductionTarget: int = 0 {target production for the assembly line}</p> <p>CurrentProduction: int = 0 {current production on the assembly line}</p> <p>Resource:</p> <ul style="list-style-type: none"> <li>Name: string = "" {name of the resource}</li> <li>Type: string = "" {type of the resource}</li> <li>HourlyProduction: int = 0 {hourly production of the resource}</li> </ul> |
| LC Report GUI               | <p>MachineDetails: string = "" {details of the machine}</p> <p>HourlyProduction: int = 0 {hourly production}</p> <p>CongestionStatus: string = "" {status of congestion}</p>  |
| Line Chief GUI              | <p>AssemblyLineLayout: AssemblyLineLayoutGUI {instance of the Assembly Line Layout GUI}</p> <p>CongestionStatus: string = "" {status of congestion}</p> <p>ReportDetails: string = "" {details of the report}</p>   |

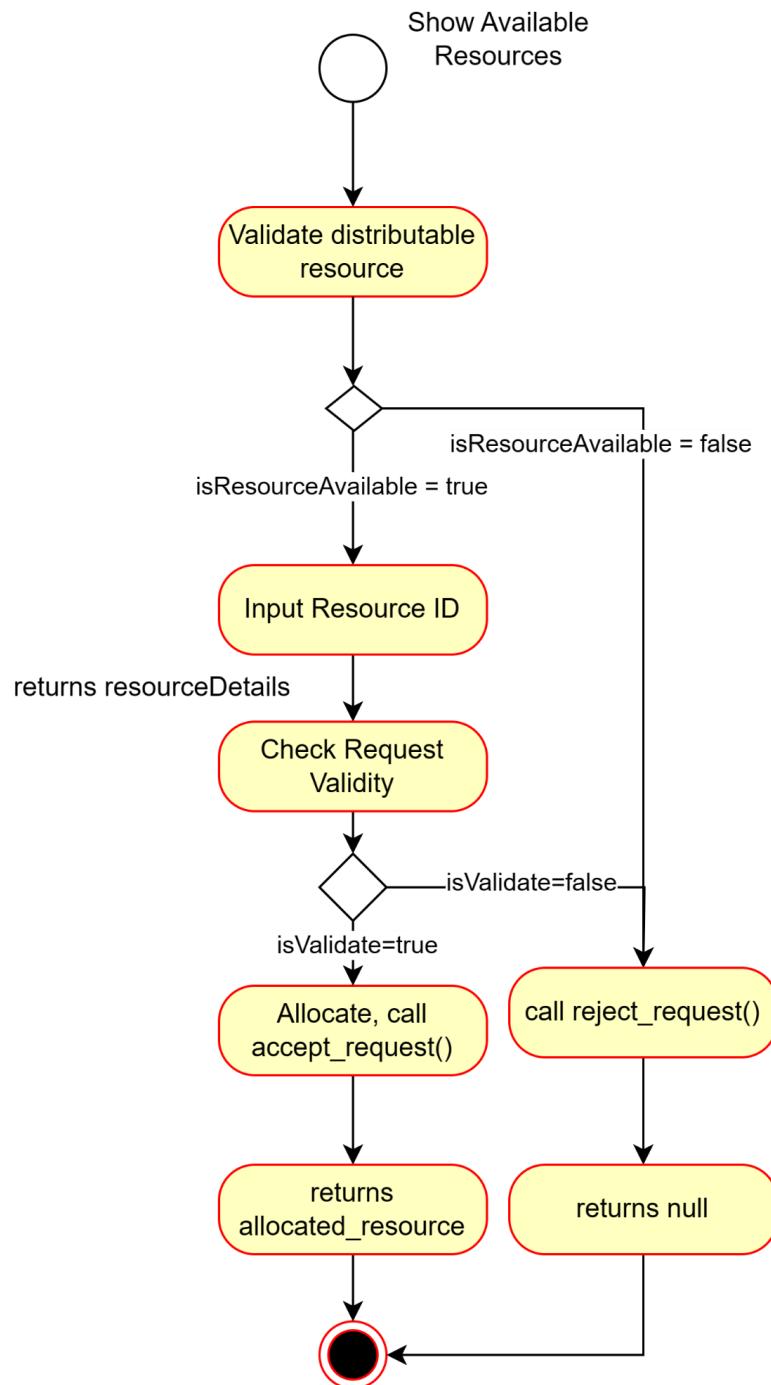
# 8. Process Flow and Pseudocodes

## 8.1 Congestion Component

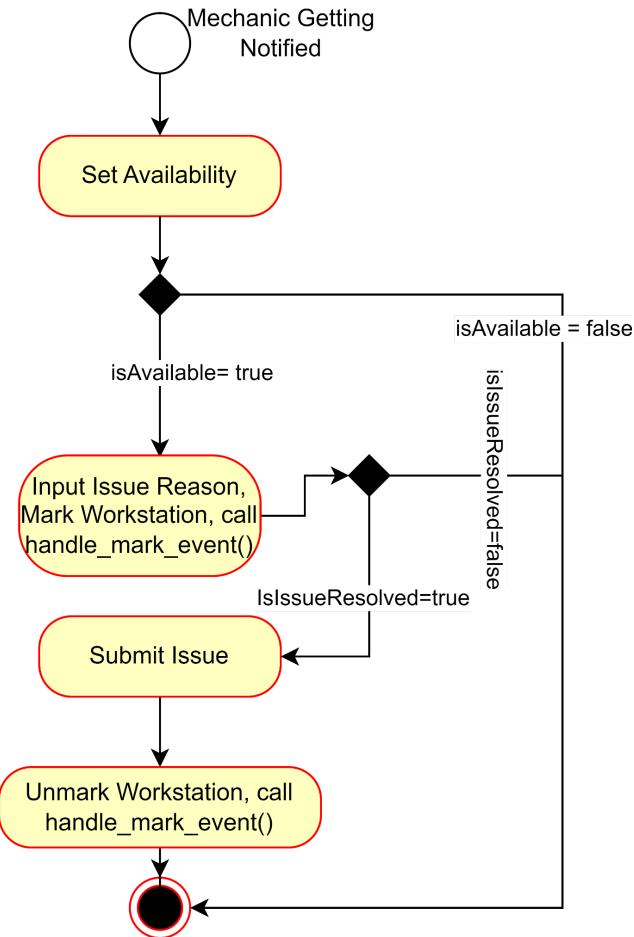
### 8.1.1 mark\_workstation()



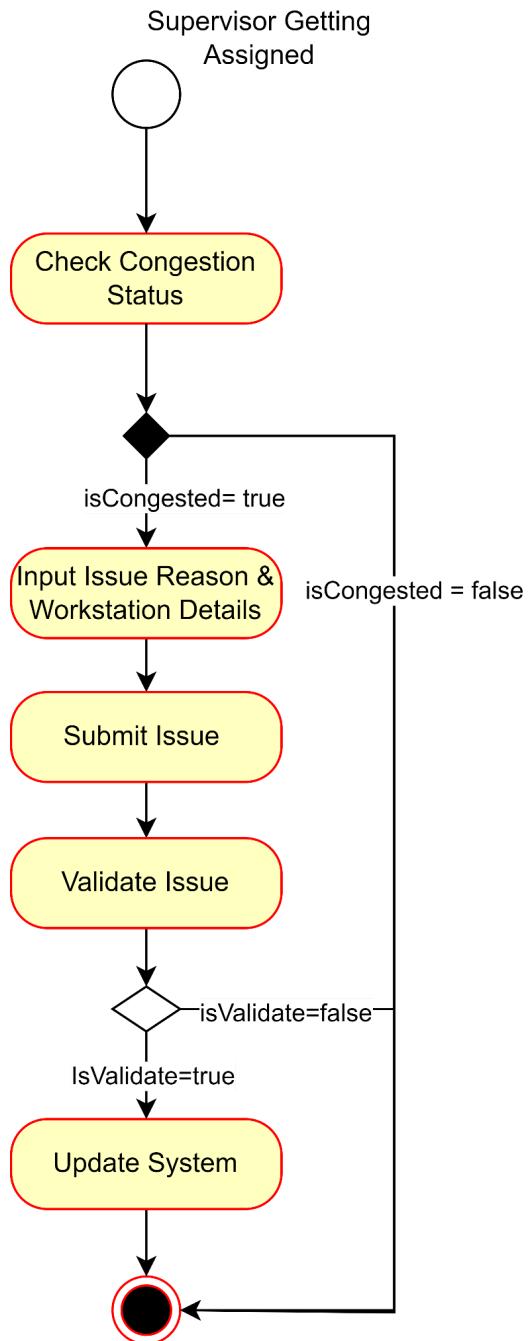
### 8.1.2 resource\_allocation()



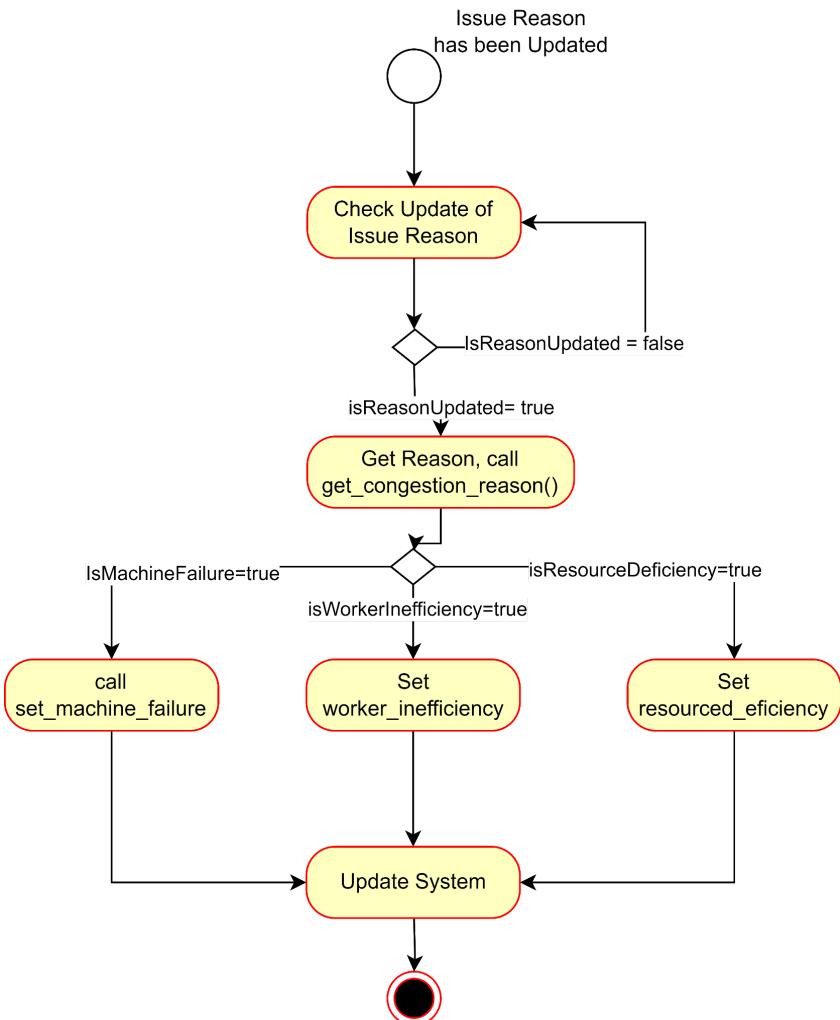
### 8.1.3 mechanic\_availability()



#### 8.1.4 record\_issue()



### 8.1.5 set\_congestion\_reason()



## 8.2 Congestion Component - ML

### 8.2.1 Class: Configuration Singleton

```
class ConfigurationSingleton:  
    private static instance: ConfigurationSingleton  
    private keyPointsTrackingPredicate: KeyPointsTrackingPredicate  
    private limit: integer  
    private referenceFrames: List<ReferenceFrame>  
  
    public static method getInstance():  
        if instance is null:  
            instance = new ConfigurationSingleton()  
        return instance
```

```

public method setKeyPointBasedTracking(predicate: KeyPointsTrackingPredicate):
    keyPointsTrackingPredicate = predicate

public method setStackLimit(limit: integer):
    this.limit = limit

public method setReferenceFrames(frames: List<ReferenceFrame>):
    referenceFrames = frames

```

### 8.2.2 Class: Preprocess Image

```

class PreprocessImageData:
    ~ imageData: ImageData
    - boundingBox: BoundingBox
    - keyPoints: List<KeyPoint>

    + preprocessImage():
        for each frame in imageData:
            generateBoundingBox(frame)
            keyPointsLocator(frame)
            dilateImage(frame)
            grayImage(frame)
            convertFramesToVectors(frame)
            convertVectorsToMatrix()

    - generateBoundingBox(frame):
        min_x = +∞
        max_x = -∞
        min_y = +∞
        max_y = -∞

        for each pixel in frame:
            if pixel is part of an object:
                x, y = pixel coordinates
                if x < min_x:
                    min_x = x
                if x > max_x:
                    max_x = x
                if y < min_y:
                    min_y = y
                if y > max_y:
                    max_y = y

        box_width = max_x - min_x
        box_height = max_y - min_y

```

```

- keyPointsLocator(frame):
key_points = detectKeyPointsInFrame(frame) // Implement key point detection for the
frame
frame.keyPoints = key_points

- dilateImage(frame):
outputFrame = createEmptyFrame(frame.width, frame.height)
structuringElement = [[1, 1, 1], [1, 1, 1], [1, 1, 1]]

for each pixel (x, y) in the input frame:
if frame(x, y) is white:
for each (i, j) in the structuring element:
    if structuringElement(i, j) is 1:
        newX = x - 1 + i
        newY = y - 1 + j
        if (newX, newY) is inside the output frame:
            outputFrame(newX, newY) = white

return outputFrame

- grayImage(frame):
outputFrame = createEmptyFrame(frame.width, frame.height)

for each pixel (x, y) in the input frame:
red, green, blue = frame(x, y).rgb
averageValue = (red + green + blue) / 3
grayscaleValue = createColor(averageValue, averageValue, averageValue)
outputFrame(x, y) = grayscaleValue

return outputFrame

- convertFramesToVectors(frame):
vector = []

for each pixel (x, y) in the frame:
append frame(x, y).value to vector

return vector

+ convertVectorsToMatrix():
matrix = []

for each vector in vectors:
append vector to matrix

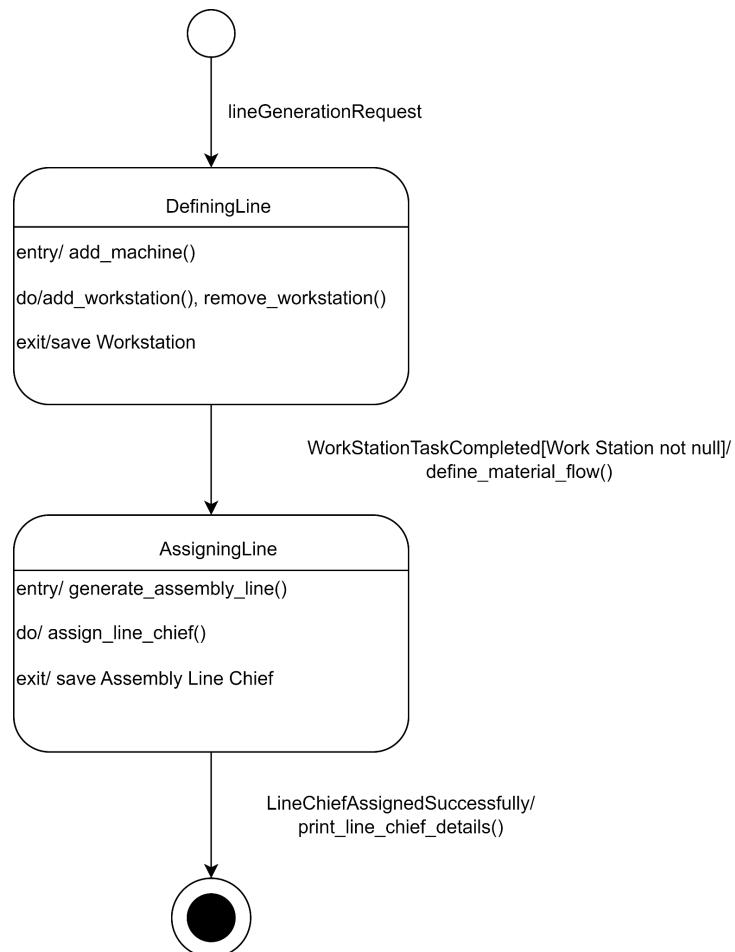
```

```
return matrix
```

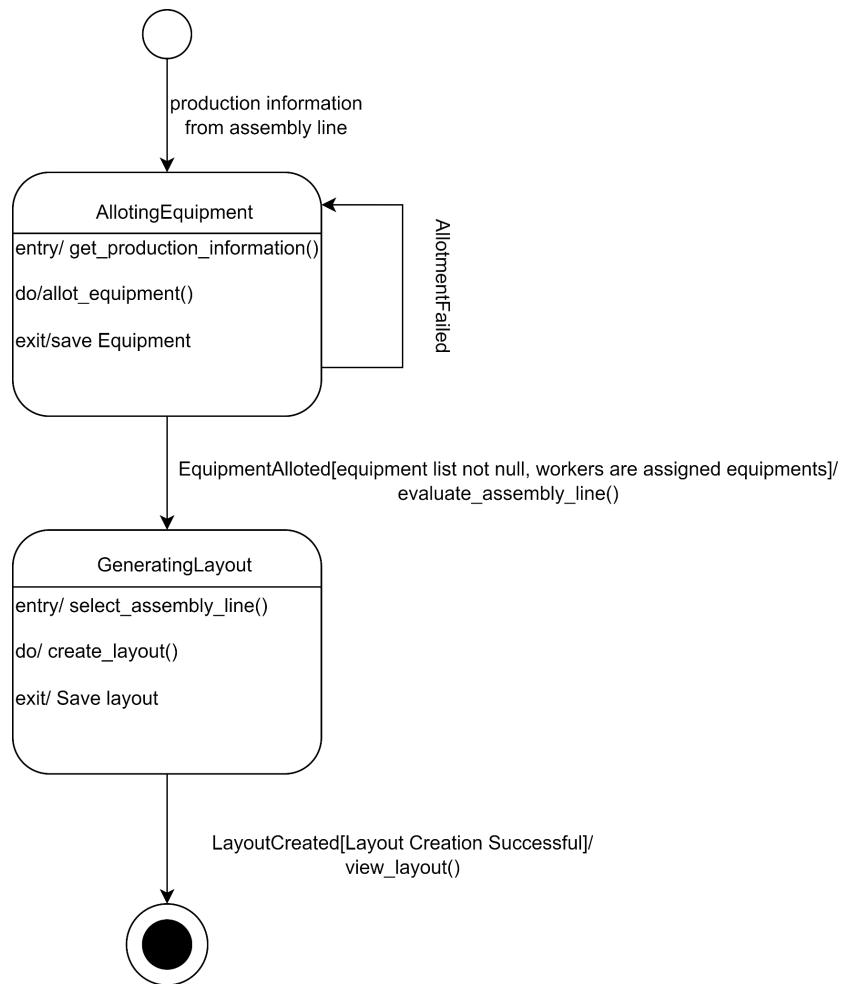
## 9. Behavioral Representations for Classes

### 9.1 Line Metrics Component

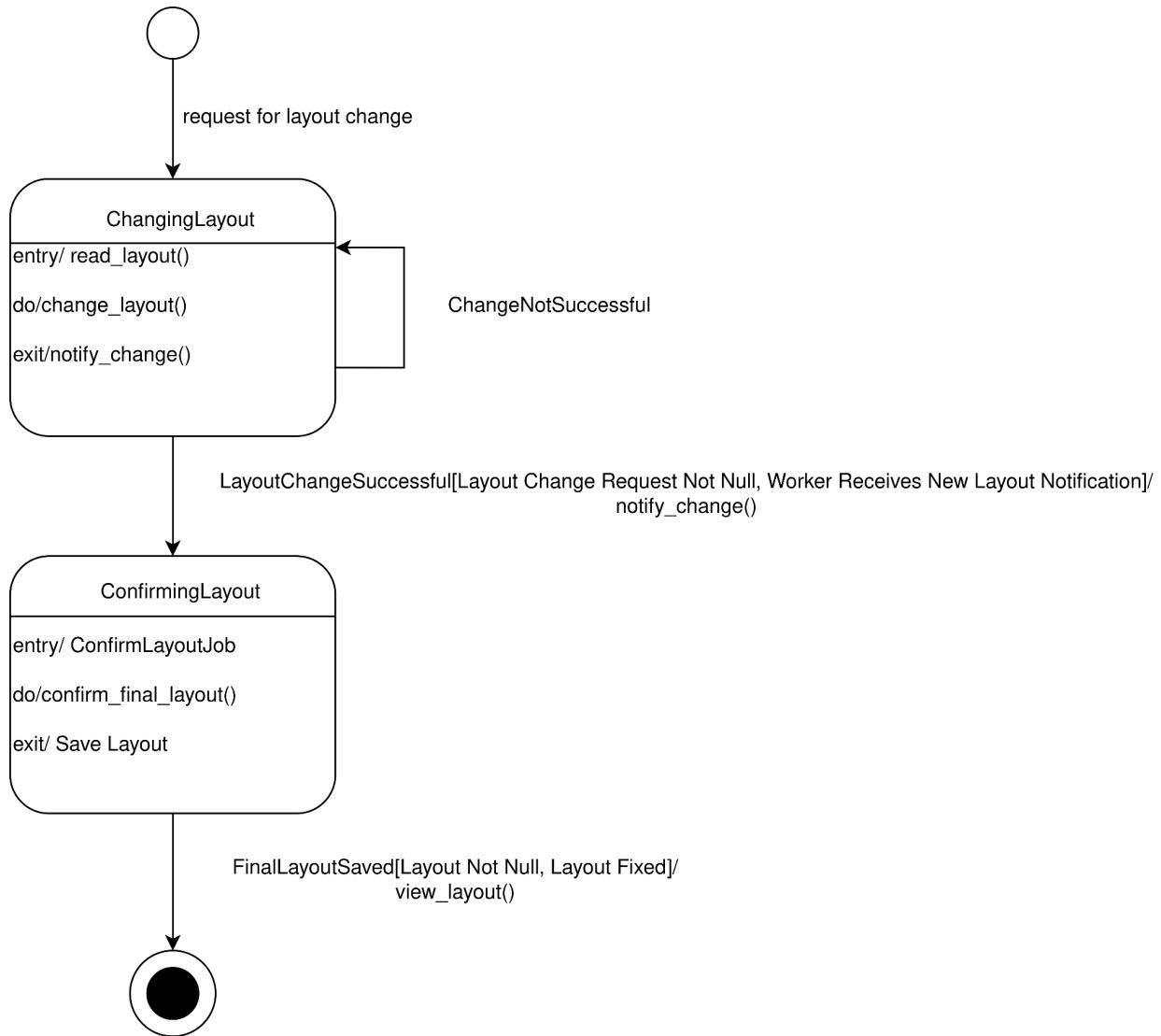
#### 9.1.1 AssemblyLine



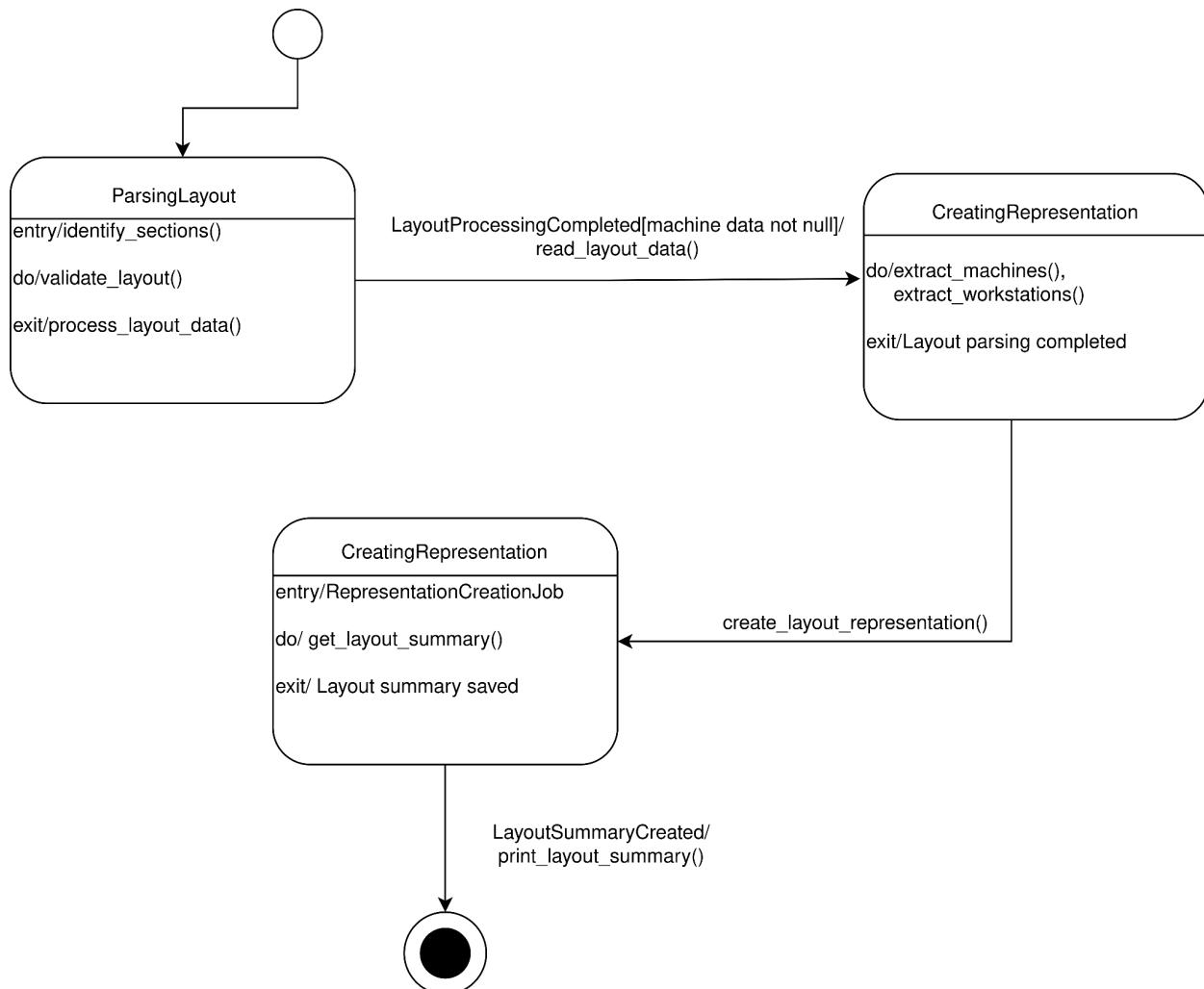
### 9.1.2 LayoutDesigner



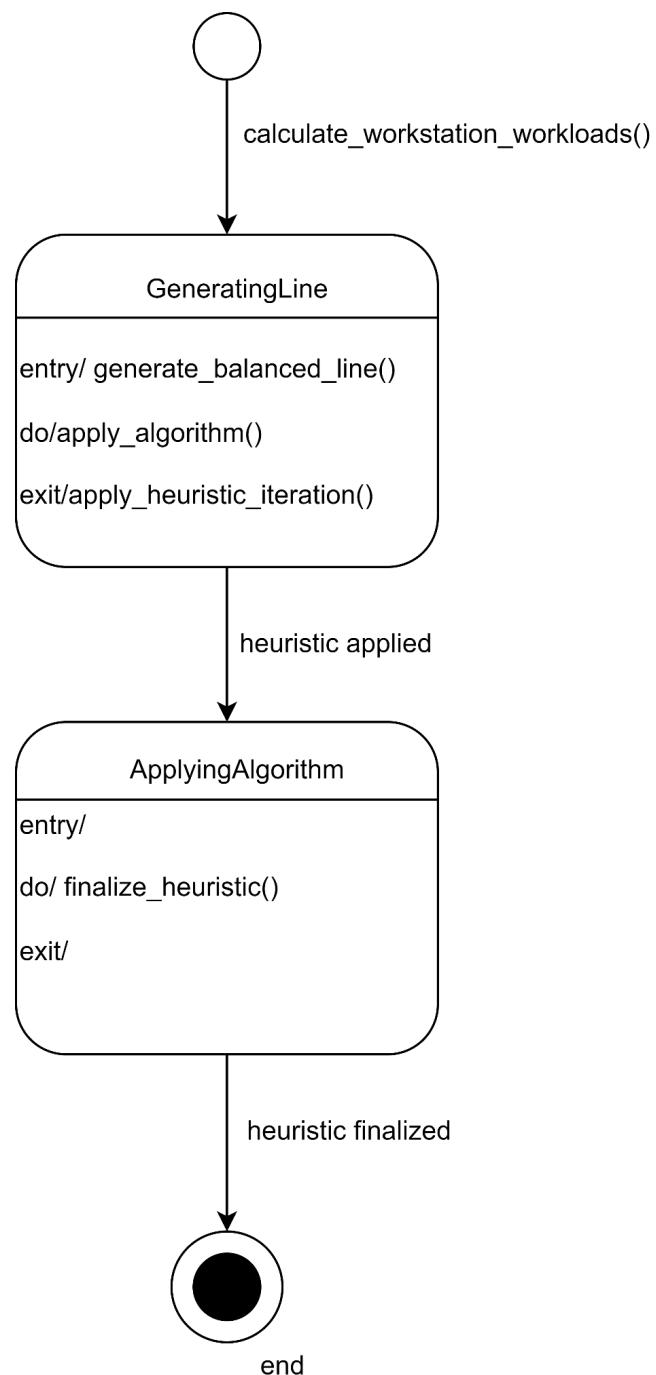
### 9.1.3 LayoutConfirmation



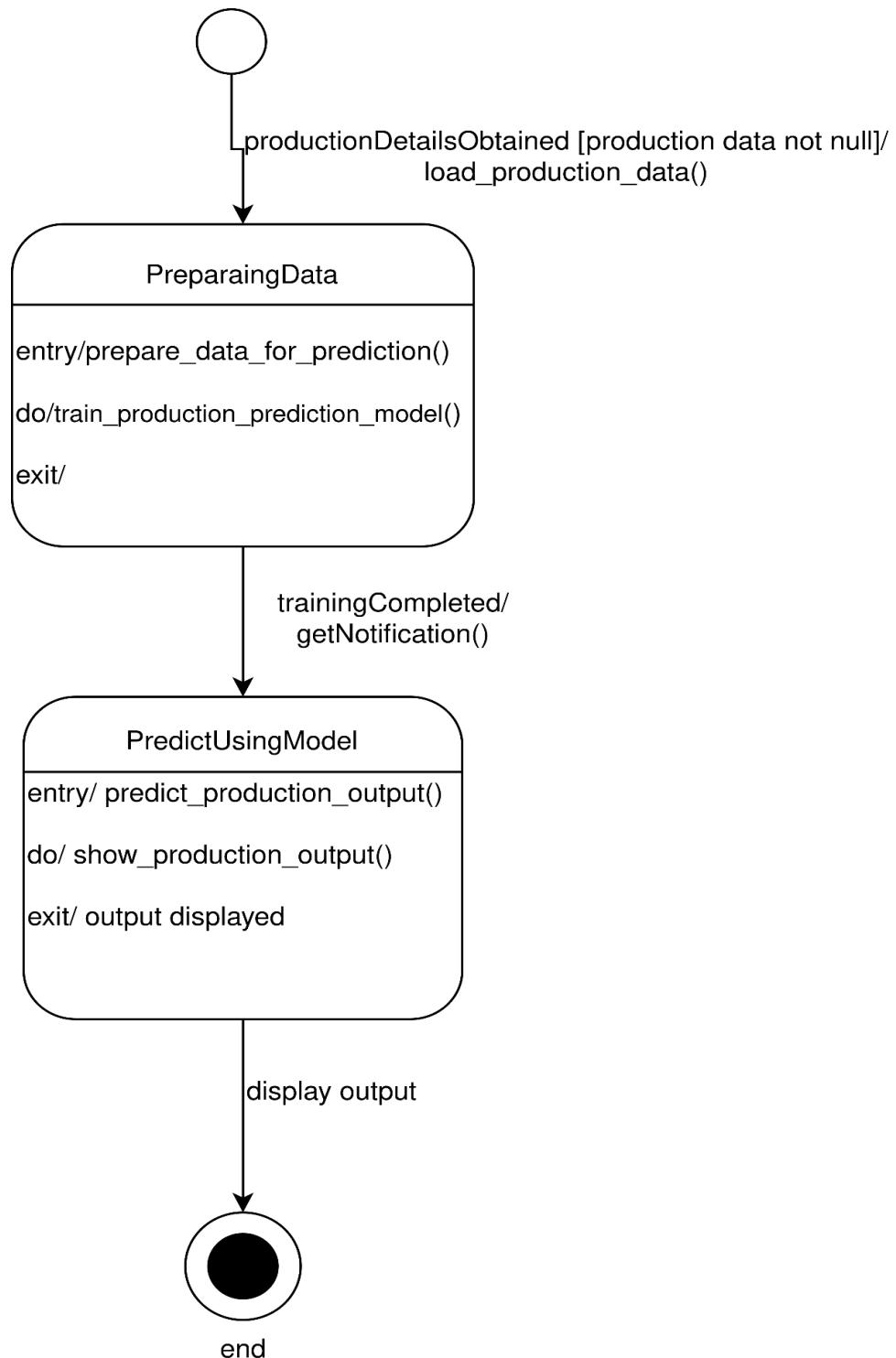
#### 9.1.4 LayoutParser



### 9.1.5 LineBalancer

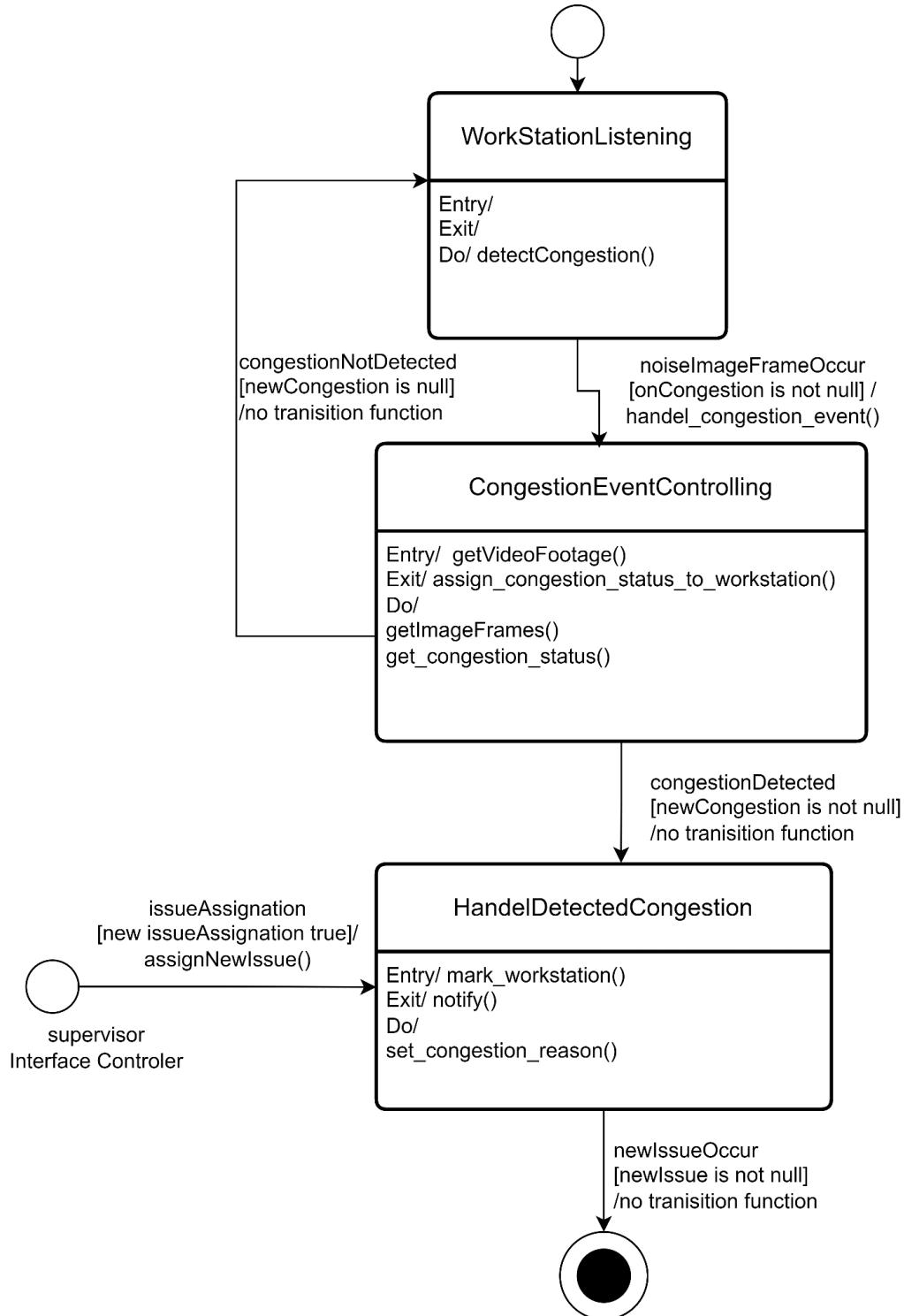


### 9.1.6 HourlyProductionPrediction

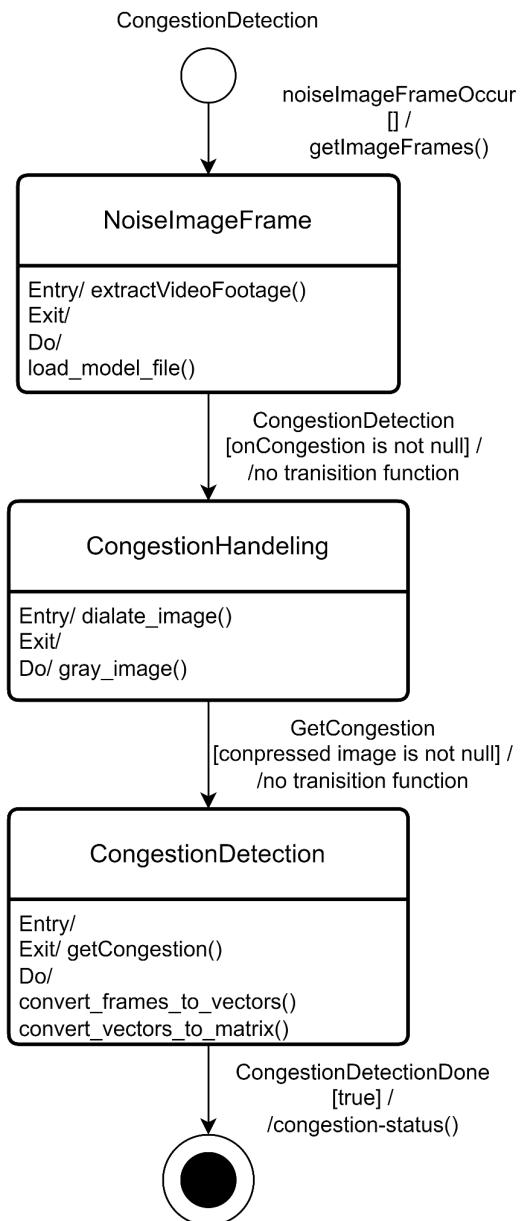


## 9.2 Congestion Component

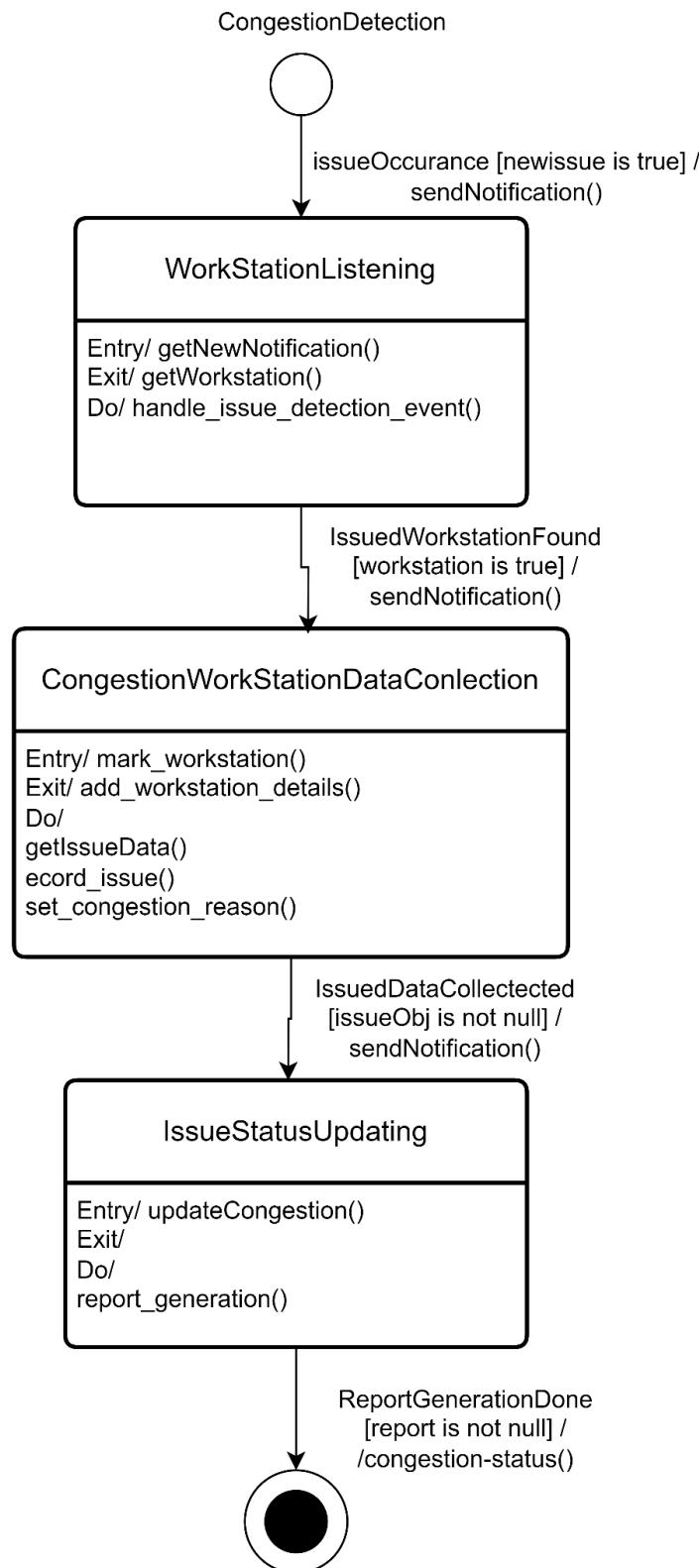
### 9.2.1 CongestionDetection



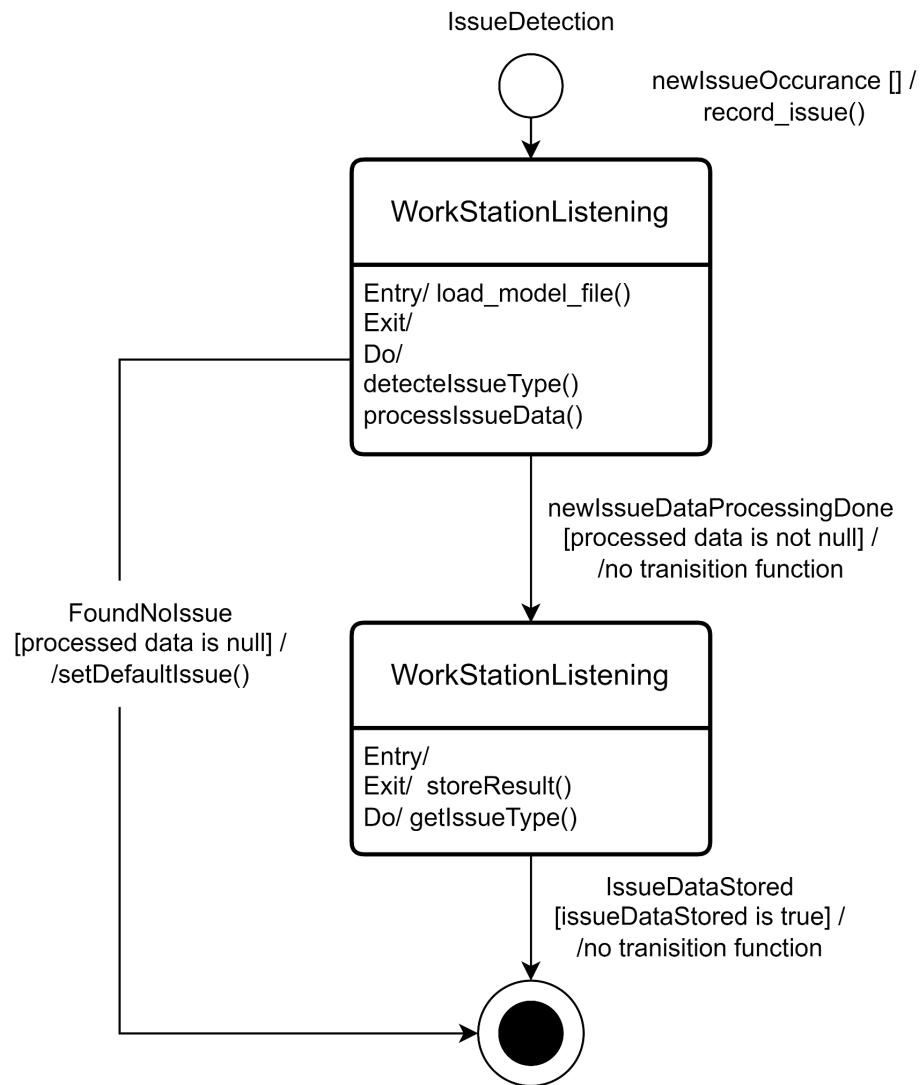
### 9.2.2 CongestionDetectionModel



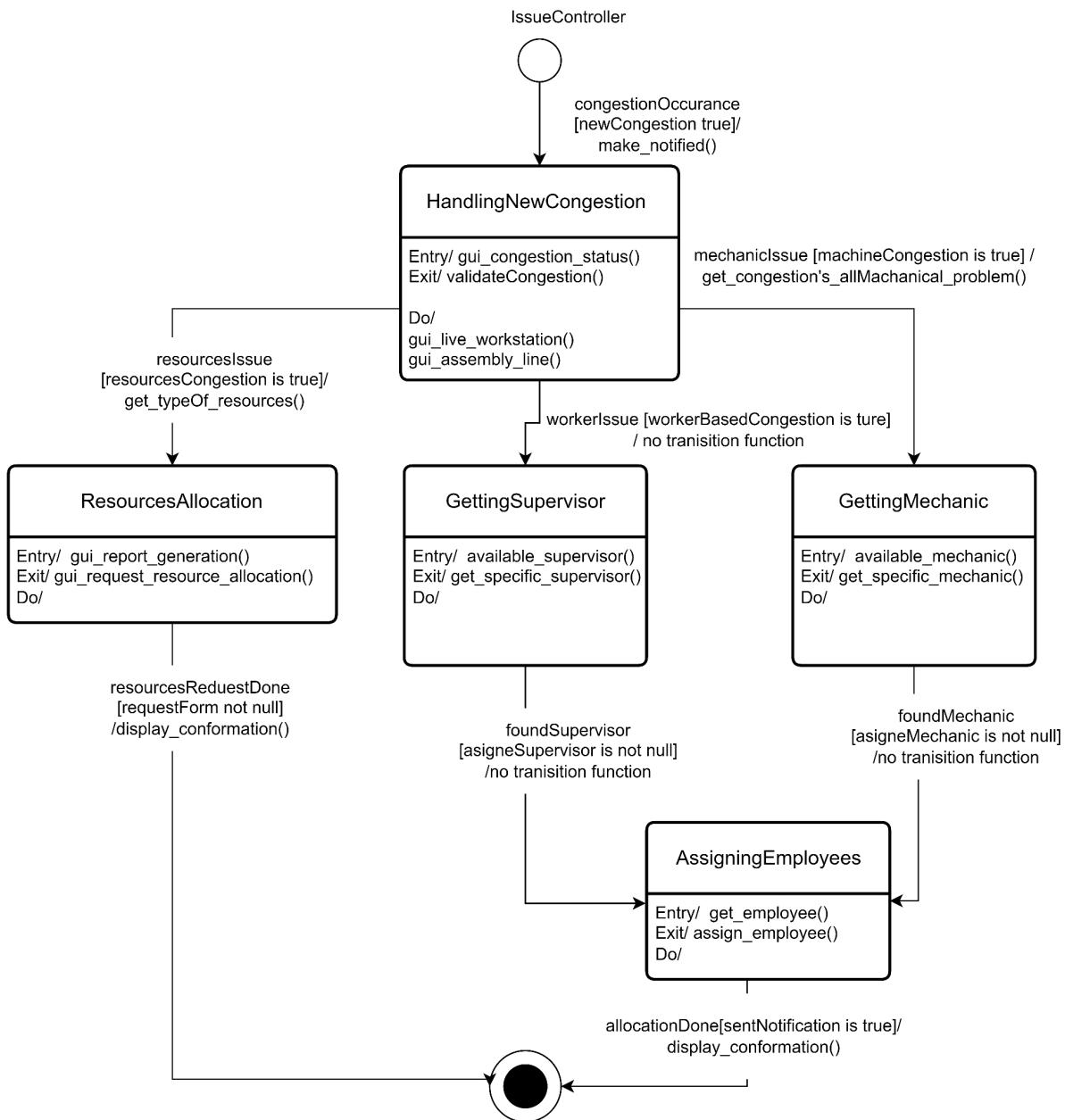
### 9.2.3 IssueDetection



#### 9.2.4 Issue Detection Model



## 9.2.5 Issue Controller



## 10. User Interface Design

### 10.1 User Analysis

The approximate ratio of males and females in the following positions have been collected from as a national average:

"Understanding the Gender Composition and Experience of Ready-Made Garment (RMG) Workers in Bangladesh, International Labor Organization (ILO), 2020"

#### 10.1.1 Supervisor

| Attribute                                 | Response   |  |              |                   |   |     |                       |          |                         |          |                        |      |
|---|--|--|--------------|-------------------|---|-----|-----------------------|----------|-------------------------|----------|------------------------|------|
| <i>Occupation</i>                         | Supervising the garment  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Level of Education</i>                 | Any, Usually SSC   |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Learning Style</i>                     | Learn from Experience  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Accessibility Preference</i>           | Mobile Phone   |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Age Range</i>                          | 20–30 Years  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Gender Profile</i>                     | Male 80%, Female 20%   |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Type of Compensation</i>               | 10~20k Taka  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Work Schedule Type</i>                 | 8AM-5PM  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Frequency of Use</i>                   | 10–15 times a day  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Primary Spoken Language</i>            | Bangla   |  |              |                   |   |     |                       |          |                         |          |                        |      |
| <i>Consequence of Mistake</i>             | <table border="1"><thead><tr><th>Mistake Type</th><th>Consequence Level</th></tr></thead><tbody><tr><td>Mistakenly Marked a workstation congested</td><td>Low</td></tr><tr><td>Recorded Wrong Reason</td><td>Very Low</td></tr><tr><td>Uploaded Wrong Tutorial</td><td>Very Low</td></tr><tr><td>Submitted Wrong Report</td><td>High</td></tr></tbody></table> |  | Mistake Type | Consequence Level | Mistakenly Marked a workstation congested | Low | Recorded Wrong Reason | Very Low | Uploaded Wrong Tutorial | Very Low | Submitted Wrong Report | High |
| Mistake Type                              | Consequence Level  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| Mistakenly Marked a workstation congested | Low  |  |              |                   |   |     |                       |          |                         |          |                        |      |
| Recorded Wrong Reason                     | Very Low   |  |              |                   |   |     |                       |          |                         |          |                        |      |
| Uploaded Wrong Tutorial                   | Very Low   |  |              |                   |   |     |                       |          |                         |          |                        |      |
| Submitted Wrong Report                    | High   |  |              |                   |   |     |                       |          |                         |          |                        |      |

|                                |   |
|--------------------------------|---|
| <i>Expertise Level</i>         | High  |
| <i>Technology Transparency</i> | Highly Required to abstract technical details |

#### 10.1.2 Line Chief

| <b>Attribute</b>                      | <b>Response</b>  |  |                     |                          |                                       |     |                     |      |                               |          |
|---------------------------------------|--|--|---------------------|--------------------------|---------------------------------------|-----|---------------------|------|-------------------------------|----------|
| <i>Occupation</i>                     | Supervising one assembly line  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Level of Education</i>             | SSC-HSC  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Learning Style</i>                 | From experience  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Accessibility Preference</i>       | Mobile Phone   |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Age Range</i>                      | 30–40 Years  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Gender Profile</i>                 | Male 75%, Female 25%   |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Type of Compensation</i>           | 30-40k Taka  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Work Schedule Type</i>             | 8AM-5PM  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Frequency of Use</i>               | Very Frequent  |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Primary Spoken Language</i>        | Bangla   |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Consequence of Mistake</i>         | <table border="1"> <thead> <tr> <th><i>Mistake Type</i></th> <th><i>Consequence Level</i></th> </tr> </thead> <tbody> <tr> <td>Assigned task to wrong supervisor etc</td> <td>Low</td> </tr> <tr> <td>Designed Bad Layout</td> <td>High</td> </tr> <tr> <td>Mistakenly Requested Resource</td> <td>Moderate</td> </tr> </tbody> </table> |  | <i>Mistake Type</i> | <i>Consequence Level</i> | Assigned task to wrong supervisor etc | Low | Designed Bad Layout | High | Mistakenly Requested Resource | Moderate |
| <i>Mistake Type</i>                   | <i>Consequence Level</i>   |  |                     |                          |                                       |     |                     |      |                               |          |
| Assigned task to wrong supervisor etc | Low  |  |                     |                          |                                       |     |                     |      |                               |          |
| Designed Bad Layout                   | High   |  |                     |                          |                                       |     |                     |      |                               |          |
| Mistakenly Requested Resource         | Moderate   |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Expertise Level</i>                | High   |  |                     |                          |                                       |     |                     |      |                               |          |
| <i>Technology Transparency</i>        | Highly Required to abstract technical details  |  |                     |                          |                                       |     |                     |      |                               |          |

### 10.1.3 Production Manager

| <b>Attribute</b>                      | <b>Response</b>  |                     |                          |                               |     |                     |      |                                       |          |
|---------------------------------------|--|---------------------|--------------------------|-------------------------------|-----|---------------------|------|---------------------------------------|----------|
| <i>Occupation</i>                     | Supervising 4-5 Assembly Lines and their Productions   |                     |                          |                               |     |                     |      |                                       |          |
| <i>Level of Education</i>             | Minimum Undergraduate  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Learning Style</i>                 | From Experience and from academic knowledge  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Accessibility Preference</i>       | Computer, Mobile Phone or any device   |                     |                          |                               |     |                     |      |                                       |          |
| <i>Age Range</i>                      | 25–45 Years  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Gender Profile</i>                 | Male 90%, Female 10%   |                     |                          |                               |     |                     |      |                                       |          |
| <i>Type of Compensation</i>           | 50-60K Taka  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Work Schedule Type</i>             | 9AM-5PM  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Frequency of Use</i>               | Moderately Frequent  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Primary Spoken Language</i>        | Bangla, English  |                     |                          |                               |     |                     |      |                                       |          |
| <i>Consequence of Mistake</i>         | <table border="1"> <thead> <tr> <th><i>Mistake Type</i></th> <th><i>Consequence Level</i></th> </tr> </thead> <tbody> <tr> <td>Mistakenly Allocated Resource</td> <td>Low</td> </tr> <tr> <td>Designed Bad Layout</td> <td>High</td> </tr> <tr> <td>Forgot to register resources properly</td> <td>Moderate</td> </tr> </tbody> </table> | <i>Mistake Type</i> | <i>Consequence Level</i> | Mistakenly Allocated Resource | Low | Designed Bad Layout | High | Forgot to register resources properly | Moderate |
| <i>Mistake Type</i>                   | <i>Consequence Level</i>   |                     |                          |                               |     |                     |      |                                       |          |
| Mistakenly Allocated Resource         | Low  |                     |                          |                               |     |                     |      |                                       |          |
| Designed Bad Layout                   | High   |                     |                          |                               |     |                     |      |                                       |          |
| Forgot to register resources properly | Moderate   |                     |                          |                               |     |                     |      |                                       |          |
| <i>Expertise Level</i>                | High   |                     |                          |                               |     |                     |      |                                       |          |
| <i>Technology Transparency</i>        | Less need for it. They are tech-savvy.   |                     |                          |                               |     |                     |      |                                       |          |

#### 10.1.4 Mechanic

| <b>Attribute</b>                | <b>Response</b>  |                     |                          |                                 |     |                     |     |                     |     |
|---------------------------------|--|---------------------|--------------------------|---------------------------------|-----|---------------------|-----|---------------------|-----|
| <i>Occupation</i>               | Fixing problematic workstations on demand  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Level of Education</i>       | Polytechnique/HSC  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Learning Style</i>           | From Experience and from academic knowledge  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Accessibility Preference</i> | Mobile Phone   |                     |                          |                                 |     |                     |     |                     |     |
| <i>Age Range</i>                | 20–30 Years  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Gender Profile</i>           | Male 95%, Female 5%  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Type of Compensation</i>     | 15-25k Taka  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Work Schedule Type</i>       | 9AM-5PM  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Frequency of Use</i>         | Few  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Primary Spoken Language</i>  | Bangla   |                     |                          |                                 |     |                     |     |                     |     |
| <i>Consequence of Mistake</i>   | <table border="1"> <thead> <tr> <th><i>Mistake Type</i></th> <th><i>Consequence Level</i></th> </tr> </thead> <tbody> <tr> <td>Accepted Wrong Machine Fix Task</td> <td>Low</td> </tr> <tr> <td>Inserted Wrong Info</td> <td>Low</td> </tr> <tr> <td>Found wrong machine</td> <td>Low</td> </tr> </tbody> </table> | <i>Mistake Type</i> | <i>Consequence Level</i> | Accepted Wrong Machine Fix Task | Low | Inserted Wrong Info | Low | Found wrong machine | Low |
| <i>Mistake Type</i>             | <i>Consequence Level</i>   |                     |                          |                                 |     |                     |     |                     |     |
| Accepted Wrong Machine Fix Task | Low  |                     |                          |                                 |     |                     |     |                     |     |
| Inserted Wrong Info             | Low  |                     |                          |                                 |     |                     |     |                     |     |
| Found wrong machine             | Low  |                     |                          |                                 |     |                     |     |                     |     |
| <i>Expertise Level</i>          | High   |                     |                          |                                 |     |                     |     |                     |     |
| <i>Technology Transparency</i>  | Less need for it. They are tech-savvy.   |                     |                          |                                 |     |                     |     |                     |     |

## 10.2 Task Analysis

### 10.2.1 User - Supervisor

#### **Task: Login**

Source: UCD Level 1.1

Subtask 1: Provide Username

Subtask 2: Provide Password

Subtask 3: Perceive Login Success Status

#### **Task: Supervisor Account Information Change**

Source: UCD Level 1.1

Subtask 1: Provide New Username if you want to change Username

Subtask 2: Provide New Description if you want to change Description

Subtask 3: Provide New Profile Picture if you want to change Profile Picture

Subtask 4: Request Password Change if you want to change Password.

#### **Task: Layout Handling**

Source: UCD Level 1.2

Subtask 1: View Live Congestion Layout

- View Live Feed of Workstation
- Marked/Congested Workstation
- View Workstation information

Subtask 2: Mark Congested Workstation Identified Manually

#### **Task: Report Generation**

UCD Level 1.3

Subtask 1: Submit Hourly Production Report

- Show Hourly Production Target
- Provide Hourly Target Reached
- Provide Additional Comments

Subtask 2: Submit Quality Report

- Provide Quality Count
- Provide Additional Comments

Subtask 3: Submit Congestion Report

- Show detected congestion count
- Provide congestion count
- Provide additional comments

### **Task: Congestion Reason**

Subtask 1: Submit Congestion Reason

- Select a congestion reason from
  - Worker inexperience
  - Faulty workstations
  - Insufficient workstations
- Provide additional comments

## 10.2.2 User - Mechanic

### **Task: Mechanic Account Information Change**

Source: UCD Level 1.1

Subtask 1: Provide New Username if you want to change Username

Subtask 2: Provide New Password if you want to change Password

Subtask 3: Provide New Description if you want to change Description

Subtask 4: Provide New Profile Picture if you want to change Profile Picture

Subtask 5: Request Password Change if you want to change Password.

### **Task: Layout Handling**

Source: UCD Level 1.2

Subtask 1: View Live Congestion Layout

- View Live Feed of Workstation
- Marked/Congested Workstation
- View Workstation information

Subtask 2: View workstations with machine issues currently

### **Task: Notification**

Subtask 1: View notification requesting machine fix task allocation

- Accept
- Reject

Subtask 2: View other Notifications

## 10.2.3 User - Line Chief

### **Task: Line Chief Account Information Change**

Source: UCD Level 1.1

Subtask 1: Provide New Username if you want to change Username  
Subtask 2: Provide New Password if you want to change Password  
Subtask 3: Provide New Description if you want to change Description  
Subtask 4: Provide New Profile Picture if you want to change Profile Picture  
Subtask 5: Request Password Change if you want to change Password.

### **Task: Resource Handling**

Source: UCD Level 1.2

Subtask 1: Request resource after viewing assembly line layout  
Subtask 2: Get congestion reason for requested resource from supervisor.

- Reject or accept the request. If rejected, notify the supervisor.
- Define the type of resource need
- Forward resource request to production manager

Subtask 3: Get notification on resource allocation.

### **Task: Layout Handling**

Source: UCD Level 1.3

Subtask 1: View assembly line layout proposed by Production Manager.  
Subtask 2: Edit assembly line layout.

- View line metrics generated for each layout.
- View available resources
- Drag a workstation over an assembly line and drop it.
- View hourly production possible for the line after dropping additional workstations.

Subtask 3: View Live Congestion Layout

- View Live Feed of Workstation
- Marked/Congested Workstation
- View Workstation information

### **Task: Report Notification**

Source: UCD Level 1.3

Subtask 1: Notification on reports that supervisor submit  
Subtask 2: Approve the reports

### **Task: Congestion Detection**

Source: UCD Level 1.4

Subtask 1: Get notification when the system detects congestion on any workstation.  
Subtask 2: Get notification when a supervisor manually marks a workstation as congested.

### **Task: Congestion Reason**

Source: UCD Level 1.4

Subtask 1: Assign a supervisor to record the reason for congestion

Subtask 2: View the reason for congestion.

- Insufficient Resource: It is handled in Task: Resource Handling
- Inexperienced Worker: Check if tutorial has been uploaded.
- Faulty Machine: Allocate a mechanic to fix it.

#### 10.2.4 User - Production Manager

##### **Task: Production Manager Account Information Change**

Source: UCD Level 1.1

Subtask 1: Provide New Username if you want to change Username

Subtask 2: Provide New Password if you want to change Password

Subtask 3: Provide New Description if you want to change Description

Subtask 4: Provide New Profile Picture if you want to change Profile Picture

Subtask 5: Request Password Change if you want to change Password.

##### **Task: Resource Handling**

Source: UCD Level 1.2

Subtask1: Get resource request from line chief.

Subtask2: View available resources.

Subtask3: Allocate resources to assembly lines.

##### **Task: Layout Handling**

Source: UCD Level 1.2

Subtask 1: Define assembly line layout.

- View line metrics generated for each layout.
- View available resources
- Drag a workstation over an assembly line and drop it.
- View hourly production possible for the line after dropping additional workstations.

Subtask 2: View assembly line layout proposed by Line Chief.

Subtask 3: View Live Congestion Layout

- View Live Feed of Workstation
- Marked/Congested Workstation
- View Workstation information

##### **Task: Report Notification**

UCD Level 1.3

Subtask 1: Get notification on report submission

Subtask 2: View Report

## 10.3 Design Choices Derived from Golden Rules

The golden rules for UI design are:

1. Place the user in control
  - a. Define interaction modes in a way that does not force a user into unnecessary or undesired actions
  - b. Provide for flexible interaction
  - c. Allow user interaction to be interruptible and undoable
  - d. Streamline interaction as skill levels advance and allow the interaction to be customized
2. Reduce the user's memory load
  - a. Reduce demand on short-term memory
  - b. Establish meaningful defaults
  - c. Define shortcuts that are intuitive
  - d. The visual layout of the interface should be based on a real-world metaphor
3. Make the interface consistent
  - a. Allow the user to put the current task into a meaningful context
  - b. Maintain consistency across a complete product line
  - c. If past interactive models have created user expectations, do not make changes unless there is a compelling reason to do so

We made the following design choices to implement the golden rules in our UI design.

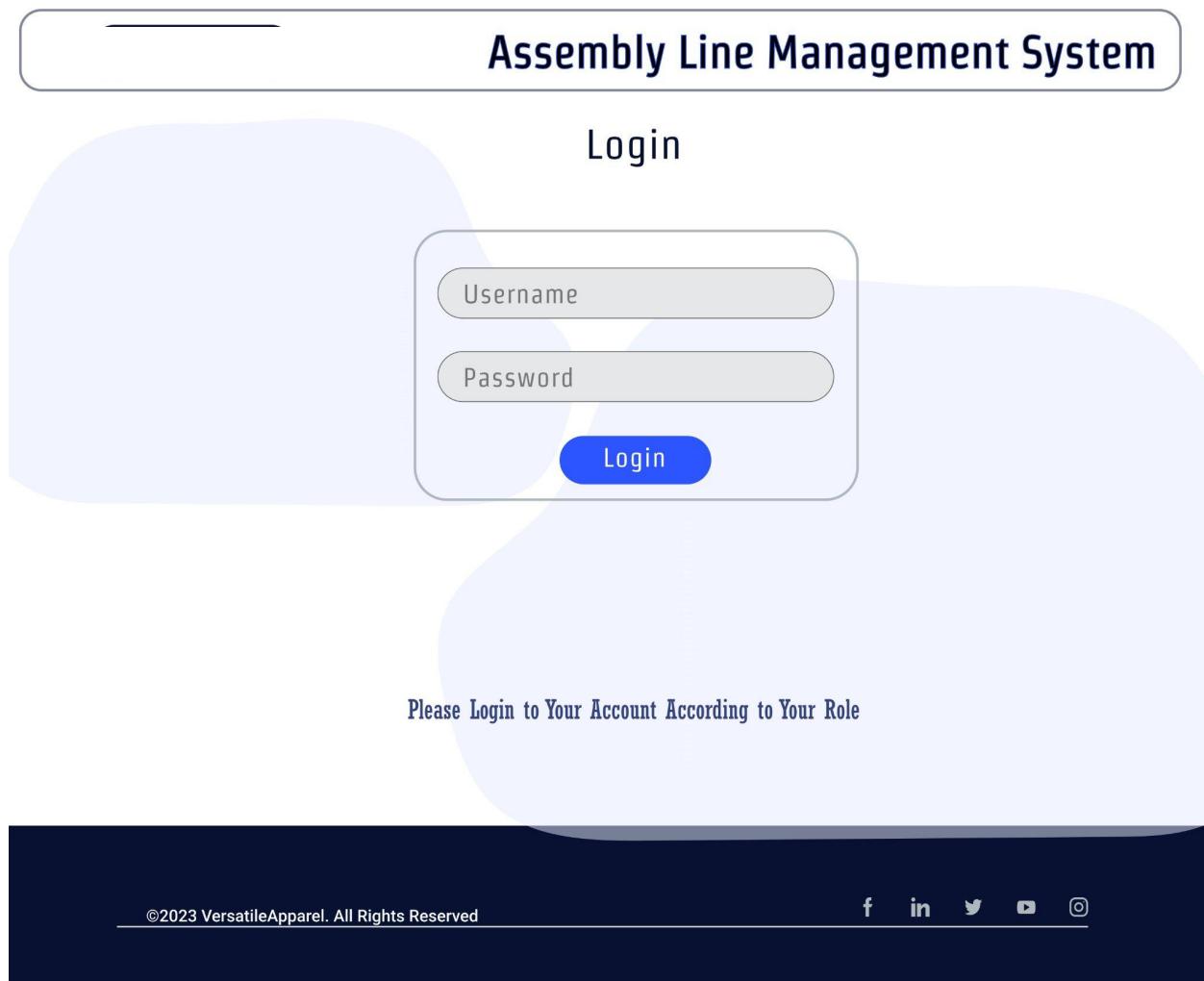
| Design Choice   | Golden Rule Id |
|---|----------------|
| All inserted data can be edited and deleted. We have edit,delete option in each table   | 1c             |
| The default values of reports' are the most occurring values.   | 2b             |
| The GUI can be interacted both from mobile phone and pc as we use websites.   | 1b             |
| We keep a side window of notification and events and logs to reduce the memory load of Line chief and production manager.       | 2a             |
| We often separate or combine tasks to ensure that to do the current task, the user doesn't need to navigate to a different GUI. | 1a             |
| The assembly line layout, drag-and-drop layout  | 2d             |
| We don't show notifications regarding report submission in the congestion notification tab.                                     | 3a             |

Everyone sees the same layout and report page.

3b

## 10.4 Prototype

A user will see this page as soon as he visits the web page.



This is the “Login” page, acting as a landing page for all users.

[Handle Layout](#)[Update Account](#)

## Update Your Account Here



Update Username

Update Password

Confirm New Password

Request Password Change

Save Changes

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

This is the “Update Account” page. All the users can access this page and edit their profile information.

# এসেম্বলি লাইন ম্যানেজম্যান্ট

লগ আউট

লেয়াউট নিয়ন্ত্রণ করুন

একাউন্ট তথ্য পরিবর্তন



MachineA17 সেলাই মেশিন

সুপারভাইজার নাম : S1

কনফার্ম

রিজেক্ট

মেশিন ১৭ এ তে জটিলা ধরা পড়েছে  
আপনাকে লাইন স্টিফ এটার কারণ অনুমতিপ্পান  
করতে বলেছে

©2023 VersatileApparel. All Rights Reserved

f in t y o

This is the view of a Mechanic account. Mechanics confirm or reject requests to repair tools. It is in Bangla as we have learned from User Analysis that the primary language of all the users of this software is Bangla. This is to provide a sample of our Bangla design.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Show Tutorial](#)[Generate Report](#)[Update Account](#)

Workstation Name

[Name 1](#)[Name 2](#)

Congestion Status

[Congested](#)[Not Congested](#)

Mark/Unmark Workstation



Machine A17 is showing Congestion. Line Chief has assigned you to solve this

This is the “Handle Layout” view of Supervisor account. It displays the congestion status of workstations.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Show Tutorial](#)[Generate Report](#)[Update Account](#)

Workstation Name

Name 1

Name 2

Congestion Status

Congested

Not Congested

Mark/Unmark Workstation



Congestion Reason!

Close

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

This is the “Handle Layout” view of Supervisor account. It displays the congestion reasons of workstations.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Show Tutorial](#)[Generate Report](#)[Update Account](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this



This is the “Show Tutorial” view of Supervisor account. It displays the recorded tutorials.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Handle Resource](#)[Generate Report](#)[Update Account](#)

Current Production: 50

Production Target: 180

Resources

Resource 1

Resource 2

Resource 3

Resource 4

Line 1

Resource 1

Resource 2

Resource 4

Line 2

Resource 3

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

---

©2023 VersatileApparel. All Rights Reserved    

This is the “Handle Layout” view of the Production manager account. It displays the resources and the allotment of those resources..

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Congestion](#)[Generate Report](#)[Update Account](#)

Production Target: 180  
Current Production: 50  
Hourly Target: (not set)

[Set Hourly Target](#)[Request Resource](#)

05/28/2023, 10:45:27 AM | Set Hourly Target

Input Target:

[Confirm](#)[Go Back](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

This is the “Handle Layout” view of Line chief account. The input target of hourly production can be sent.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Congestion](#)[Generate Report](#)[Update Account](#)

Production Target: 180  
Current Production: 50  
Hourly Target: (not set)

[Set Hourly Target](#)[Request Resource](#)

## Request Resource

Type Of Machine:

Number Of Machine:

[Request](#)[Go Back](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

This is the “Handle Layout” view of Line chief account. The requests for resources including its type and number can be sent.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Congestion](#)[Approve Report](#)[Update Account](#)

Production Target: 180  
Current Production: 50  
Hourly Target: (not set)

[Set Hourly Target](#)[Request Resource](#)

| Model   | Machine Type | Hourly Production | Congestion Status | Marked? | Live Camera (Optional)      |
|---------|--------------|-------------------|-------------------|---------|-----------------------------|
| Model 1 | Type 1       | 11                | Congested         | Yes     | No Camera                   |
| Model 2 | Type 2       | 7                 | Not Congested     | No      | <a href="#">Open Camera</a> |
| Model 3 | Type 3       | 5                 | Not Congested     | No      | <a href="#">Open Camera</a> |

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

©2023 VersatileApparel. All Rights Reserved

[f](#) [in](#) [t](#) [y](#) [o](#)

This is the general “Handle Layout” view of Line chief account, displaying all the details.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Congestion](#)[Approve Report](#)[Update Account](#)

Production Target: 180  
Current Production: 50  
Hourly Target: (not set)

[Set Hourly Target](#)[Request Resource](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

| Model   | Machine Type | Hourly Production | Congestion Status | Marked? | Live Camera (Optional) |
|---------|--------------|-------------------|-------------------|---------|------------------------|
| Model 1 | Type 1       | 11                | Congested         | Yes     | No Camera              |
| Model 2 | Type 2       | 7                 | Not Congested     |         |                        |
| Model 3 | Type 3       | 5                 | Not Congested     |         |                        |

## Assign Supervisor

[Submit](#)[Close](#)

©2023 VersatileApparel. All Rights Reserved

f in tw yt @

This is the general “Handle Layout” view of Line chief account, where Supervisor is assigned according to the requirement of the congestion reason.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Congestion](#)[Approve Report](#)[Update Account](#)

Production Target: 180  
Current Production: 50  
Hourly Target: (not set)

[Set Hourly Target](#)[Request Resource](#)

| Model   | Machine Type | Hourly Production | Congestion Status | Marked? | Live Camera (Optional) |
|---------|--------------|-------------------|-------------------|---------|------------------------|
| Model 1 | Type 1       | 11                | Congested         | Yes     | No Camera              |
| Model 2 | Type 2       | 7                 | Not Congested     | No      | Camera                 |
| Model 3 | Type 3       | 5                 | Not Congested     | No      | Camera                 |

**Machine Failure!**

[Request Mechanic](#)[Close](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

©2023 VersatileApparel. All Rights Reserved

f in tw yt @

This is the general “Handle Layout” view of Line chief account, where new machines can be requested upon machine failure according to the requirement of the congestion reason.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Congestion](#)[Approve Report](#)[Update Account](#)

Production Target: 180  
Current Production: 50  
Hourly Target: (not set)

[Set Hourly Target](#)[Request Resource](#)

| Model   | Machine Type | Hourly Production | Congestion Status | Marked? | Live Camera (Optional) |
|---------|--------------|-------------------|-------------------|---------|------------------------|
| Model 1 | Type 1       | 11                | Congested         | Yes     | No Camera              |
| Model 2 | Type 2       | 7                 | Not Congested     |         |                        |
| Model 3 | Type 3       | 5                 | Not Congested     |         |                        |

Inexperienced Worker.  
Supervisor is showing tutorial!

[Close](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

This is the general “Handle Layout” view for Line Chief account where Supervisor is showing tutorial according to the requirement of the congestion reason.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Handle Resource](#)[Generate Report](#)[Update Account](#)

Congestion Issue Description:

Workstation Number:

Comments:

[Submit Congestion Issue](#)

Hourly Target Report

Quality Report

[Congestion Report](#)Machine A17 is showing  
Congestion. Line Chief has  
assigned you to solve this

This is the interface of “Generate Report” for the Line chief account where congestion report can be created.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Handle Resource](#)[Generate Report](#) ▾[Update Account](#)

## Hourly Target Report

Hourly Target Reached So Far:

0

Unit of Production:

(e.g. Number of Completed Garments)

Comments:

(e.g. This production has been good this time)

[Submit Production Count](#)

[Quality Report](#)

[Congestion Report](#)

Machine A17 is showing Congestion. Line Chief has assigned you to solve this



This is the interface of “Generate Report” for the Line chief account where hourly target report is being created.

# Assembly Line Management System

[Logout](#)[Handle Layout](#)[Handle Resource](#)[Generate Report](#)[Update Account](#)

Units of Products Checked:

0

Units of Defected Products Found:

0

Unit of Production:

(e.g. Number of Completed Garments)

Comments:

(e.g. Some Quality Issues Occurred)

[Submit Quality Inspection Result](#)

Hourly Target Report

[Quality Report](#)

Congestion Report

Machine A17 is showing Congestion. Line Chief has assigned you to solve this



This is the interface of “Generate Report” for the Line chief account where quality report is being created.

# Assembly Line Management System

Logout



| Report Id | Generated At            | Download                  |
|-----------|-------------------------|---------------------------|
| 01        | 05/13/2023, 08:25:32 PM | <button>Download</button> |
| 02        | 05/28/2023, 10:45:27 AM | <button>Download</button> |
| 03        | 05/21/2023, 02:14:57 PM | <button>Download</button> |

Machine A17 is showing Congestion. Line Chief has assigned you to solve this

©2023 VersatileApparel. All Rights Reserved

f in t y o

This is the “View Report” interface for all type user accounts.

Prototype of the UI of our project:

[https://www.figma.com/file/yWDCxtcF3Qidhe61U7cCTh/Figma-Website-Template---Landing-Page-\(Free\)-\(Community\)?type=design&node-id=0%3A88&mode=design&t=JYNByafaRoRKK4EZ-1](https://www.figma.com/file/yWDCxtcF3Qidhe61U7cCTh/Figma-Website-Template---Landing-Page-(Free)-(Community)?type=design&node-id=0%3A88&mode=design&t=JYNByafaRoRKK4EZ-1)

## 10.4.1 Video Demonstration

The video link for overview of the UI:

<https://drive.google.com/file/d/15skT9KBHngv1TLNnHRjLpj2CZp3Pkv5w/view?usp=sharing>

# 11. Elaborating Deployment Diagram

First, a deployment diagram is built. Then at a later phase, it is refined into instance form - that is, instantiating each deployment unit with created artifacts etc. According to UML2.0 Deployment specification [1] and Pressman(Page 248-249).

## 11.1 Deployment Diagram

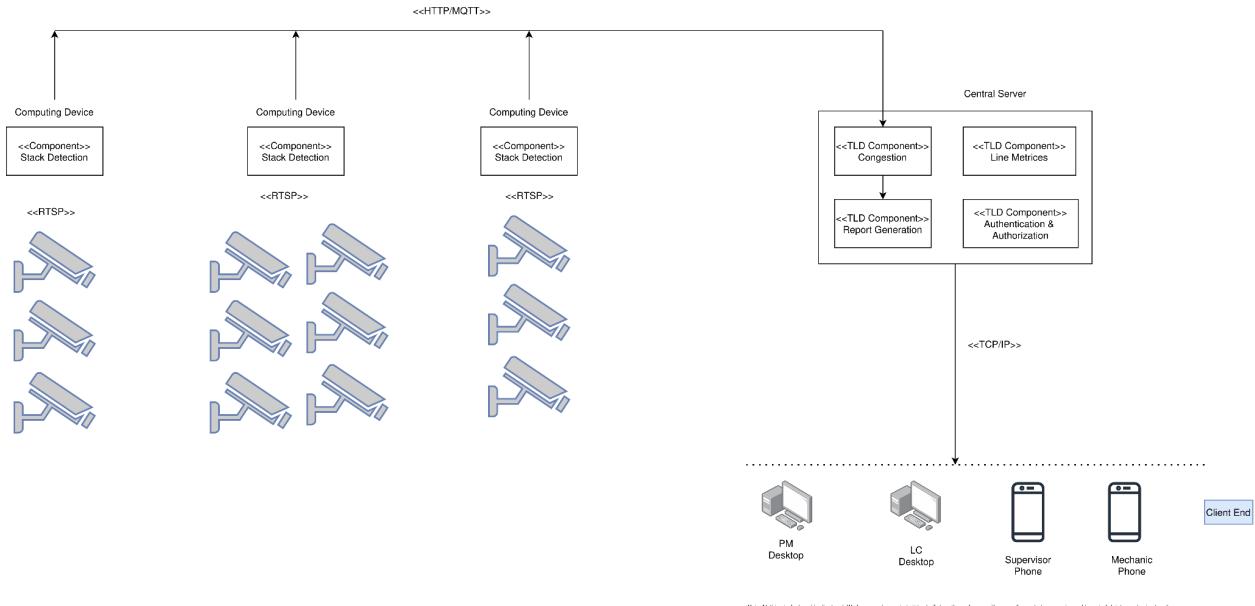


Fig: General Overview of Deployment Scenario

Note that, we use the syntax <<TLD Component>> to indicate components from Top Level Diagram(TLD) in our design document. And we use <<Component>> to define the parts of the TLD Components, refined after applying patterns to TLD Components in section 3.

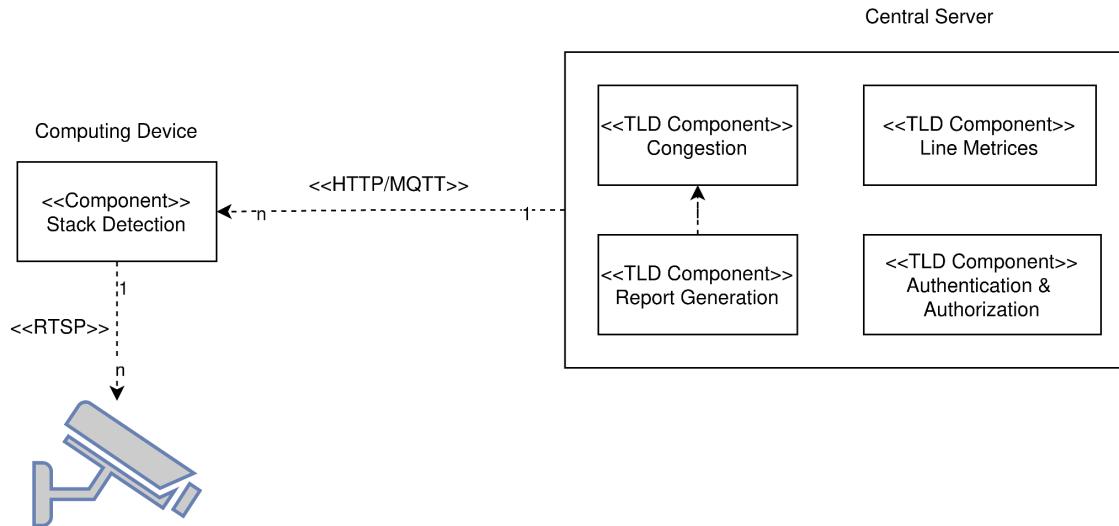


Fig: Deployment Diagram

**RTSP (Real-Time Streaming Protocol):** A network control protocol designed for use in entertainment and communication systems to control streaming media servers.

**MQTT (Message Queuing Telemetry Transport) protocol:** MQTT is a lightweight and efficient publish-subscribe messaging protocol designed for low-bandwidth, high-latency, or unreliable networks, which makes it well-suited for IoT applications.

## 11.2 Instance Form of Deployment Diagram

UML2.0 Syntax was developed for this purpose [1] and any use of it outside this scenario is voluntary and not standard.

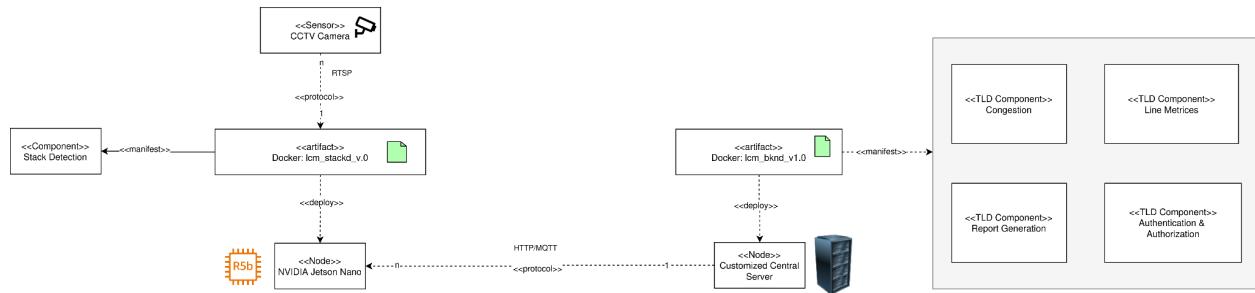


Fig: Instantiating Components into Artifacts and Deploying them [2]

## 11.3 Reference

[1]: <https://www.uml-diagrams.org/deployment-diagrams.html>

[2]:

<https://www.uml-diagrams.org/web-application-uml-deployment-diagram-example.html?context=depl-examples>