|  |  |
| --- | --- |
| Garments inventory system |  |
|  |  |
| 2/11/2014 | Group-5 |
|  | Course no: CSE 308 course name: information system design  Submitted by: 1005047,51,53,56 &57 |

Garments inventory system

Group-5

Introduction:

Economy of Bangladesh is largely dependent on the garments sector. It has attracted worldwide attention in the sector of garments. Garments accounts for a nominal 76% of the country’s exports. International quality compliances are made at every stage. Here about 1.8 million workers are engaged. Bangladesh is placed 12 among the garments manufacturing countries in the world. But this sector still has a lot of untapped potential. Moreover, this sector is still using aged old traditional system for all its tasks. There are lots of scopes to automate many of its tasks which will not only increase their productivity but also will help them to take decisions in no times. So this sector really demands automated in its own rights.

Our System:

Our proposed system is “GARMENTS INVENTORY SYSTEM”. It is a fully automated system which will keep track of the products and stocks of a garments industry. At present, garment industries are using ancient paper-based methods to maintain their stock and accounts record. They keep their records in register books and even calculation is done manually sometimes. This process is error-prone, lengthy and cumbersome. By implementing our system we can save time, minimize losses and avoid mistakes.

Our subsystems:

Our system is divided into 4 subsystems in total. They are listed below:

1. Raw material storage
2. Raw material forwarding
3. Production stage tracking
4. Delivery subsystem

1. Raw material storage:

This subsystem deals with the storage of raw material which is needed for production. Common raw materials needed for garments industry are cotton and unstitched clothes of various types and colors. This part of our inventory system automates the storage of these raw materials. The system automatically keeps track of the inventory balance. So when a demand comes it can be easily said whether there are enough raw materials.

Actors for this subsystem:

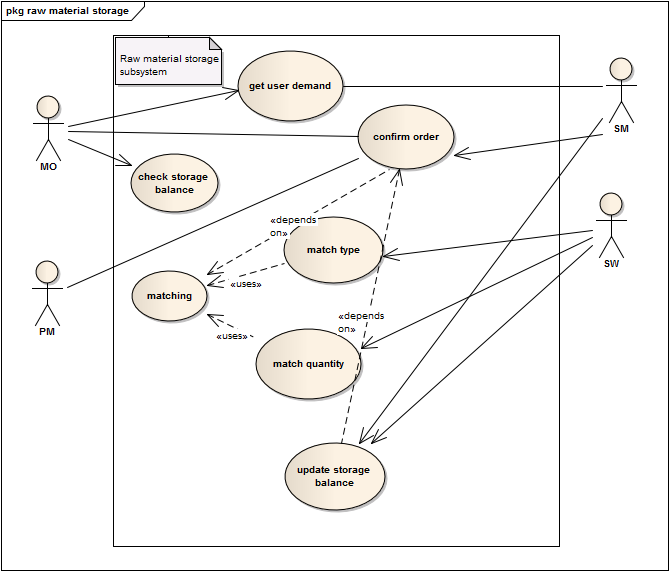
1. Main office
2. Storage manager
3. Storage worker
4. Production manager

Actors Glossary:

|  |  |  |
| --- | --- | --- |
| ACTOR | SHORT KEY | ACTIVITY SCOPE |
| Main office | MO | Places a demand for raw material, gives confirmation on receiving materials |
| Storage manager | SM | Checks and updates the storage balance |
| Storage worker | SW | Inputs the products in storage |
| Production manager | PM | Starts production on receiving materials |

Use case glossary:

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Use case id | Description | Participant actors and roles |
| Get user demand | 1.1 | filling up a demand form depending on the nature of the contract | MO fills the form, SM receives it. |
| Match type | 1.2 | After receiving the demands, the type of the products is matched | SW matches the requirements |
| Match quantity | 1.3 | After receiving the demands ,the quantity is matched | Same as type matching |
| Confirm order | 1.4 | After matching the order is confirmed | SM confirms the order, sends the goods to the factory, PM receives it, starts production, MO receives confirmation |
| Update storage balance | 1.5 | After sending an order, the balance of storage is updated | SW updates the balance, SM oversees it for transparency. |
| Check storage balance | 1.6 | Checking the current balance of the inventory at any time. | MO checks the balance, system provides output. |

Use case diagram: 

1.1: GET USER DEMAND

|  |  |
| --- | --- |
| Use case name: | Get user demand |
| Use case id | 1.1 |
| Priority | High |
| Primary system actor | Main office(MO) |
| External receiver actor | Storage manager(SM) |
| Description | filling up a demand form depending on the nature of the contract |
| Trigger by | MO |

Table: Use Case Narrative for get user demand

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: fill up a demand form and submit it online | Step 2: forward the form to the storage manager |
|  | Step 3: make an entry |

Documentation of the events of the get user demand use case:

Conclusion: concludes when an entry of the demand is made.

Post condition: notify the main office of receiving their demand.

Implementation issues: a GUI will be provided for the MO to fill up the demand form online.

1.2 MATCH TYPE

|  |  |
| --- | --- |
| Use case name: | Match type |
| Use case id | 1.2 |
| Priority | High |
| Primary business actor | Storage worker(SW) |
| Description | Matching the type of products in the inventory according to the order form |
| Trigger by | SW |

Table: use case narrative for match type

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step1: SW requests the demand form | Step 2: show the submitted demand form |
| Step 3: SW matches the type of material demanded to the type of material stored. | Step 4: make an entry (whether match found or not found) in the system. |

Alternate course of events:

|  |  |
| --- | --- |
| Actor action | System response |
|  |  |
|  | Step 4: make an entry of match not found in the system. |
| Step 5:notify main office |  |
| Step 6: main office buys necessary materials. |  |

Documentation of the events of the match type use case:

Conclusion: concludes when an entry of the match found is made.

Post condition: notify the main office of finding a match of their demand.

Implementation issues: system will show the details of the materials currently in the storage.

1.3 MATCH QUANTITY

|  |  |
| --- | --- |
| Use case name: | match quantity |
| Use case id | 1.3 |
| Priority | High |
| Primary system actor | Storage worker(SW) |
| Description | Matching the quantity of products in the inventory according to the order form |
| Trigger by | SW |

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step1: SW requests the demand form | Step 2: show the submitted demand form |
| Step 3: SW matches the quantity of material demanded to the Quantity of material stored. | Step 4: make an entry (whether match found or not found) in the system. |

Alternate course of events:

|  |  |
| --- | --- |
| Actor action | System response |
|  |  |
|  | Step 4: make an entry of match not found in the system. |
| Step 5:notify main office |  |
| Step 6: main office buys necessary materials. |  |

Documentation of the events of the match quantity use case:

Conclusion: concludes when an entry of the match found is made.

Post condition: notify the main office of finding a match of their demand.

Implementation issues: system will show the details of the materials currently in the storage.

1.4 CONFIRM ORDER

|  |  |
| --- | --- |
| Use case name: | Confirm order |
| Use case id | 1.4 |
| Priority | High |
| Primary system actor | Storage manager(SM) |
| Other participating actors | Main office(external receiver actor),production manager(external server actor) |
| Description | Confirming the order, notifying the production manager and main office |
| Trigger by | SM |

Table: use case narrative for Confirm order

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
|  | Step 1: system notifies the SM of finding a match with the order form |
| Step 2: SM rechecks the order form and storage materials(optional) |  |
| Step 3:SM requests to send a confirmation to MO and PM | Step 4: system notifies MO and PM of order confirmation |

Documentation of the events of the Confirm order use case:

Conclusion: concludes when an notification is sent to MO and PM

Pre-condition: match has to be found in type matching and quantity matching use case.

Post condition: PM will start production.

Implementation issues: system will send the notification to the MO and PM.

1.5 UPDATE STORAGE BALANCE:

|  |  |
| --- | --- |
| Use case name: | Update storage balance |
| Use case id | 1.5 |
| Priority | High |
| Primary system actor | Storage worker(SW) |
| Other participating actor | Storage manager(SM) |
| Description | Updating the inventory balance of storage after confirming an order |
| Trigger by | SM |

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: SW requests to change the storage balance | Step 2: system provides a form for changing balance |
| Step 2: SW inputs the amount of material sent | Step 3: system subtracts the amount of material sent from inventory. |
|  | Step 4:system shows the change made to SW |
|  | Step 5:system notifies the SM of changes made |
| Step 6: SM approves of the changes | Step 7: system saves the changes. |

Documentation of the events of the update storage balance use case:

Conclusion: concludes when an entry of the change is made

Pre-condition: order has to be confirmed.

Post condition: storage balance will be updated.

Implementation issues: system will provide a form for SW, send the notification to the SM and calculate current balance.

1.6 CHECK STORAGE BALANCE

|  |  |
| --- | --- |
| Use case name: | Check storage balance |
| Use case id | 1.6 |
| Priority | High |
| Primary business actor | MO |
| Description | checking the inventory balance |
| Trigger by | MO |

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: MO requests to check the storage balance | Step 2: system provides detail of current balance |

Documentation of the events of the check storage balance use case:

Conclusion: concludes when detail has been shown to the MO

Implementation issues: system will provide a GUI for MO.

2. Raw material forwarding:

This subsystem deals the forwarding of raw material from storage to factory. When the raw materials are sent from the factory the production manager and main office both receive a confirmation. Then they check the status of the raw materials. After that the floor wise target for each floor is set.

Actors for this subsystem:

1. Main office
2. Raw material storage subsystem
3. Production manager

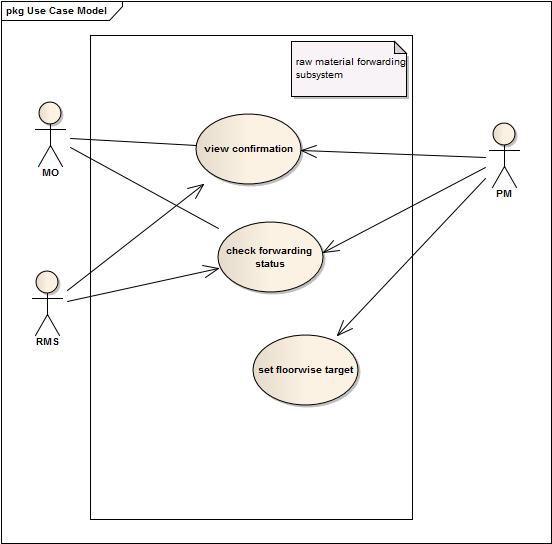
Actors Glossary:

|  |  |  |
| --- | --- | --- |
| ACTOR | SHORT KEY | ACTIVITY SCOPE |
| Main office | MO | Checks the status of raw material forwarding from storage to factory, receives confirmation of raw material delivery. |
| Raw material storage subsystem | RMS | Gives confirmation of sending the raw material from storage |
| Production manager | PM | Receives confirmation of raw material delivery, Checks the status of raw material forwarding from storage to factory, sets floor wise production target. |

Use case glossary:

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Use case id | Description | Participant actors and roles |
| View confirmation | 2.1 | Showing confirmation of sending raw materials | RMS sends confirmation, MO and PM receive it |
| Check forwarding status | 2.2 | After receiving the confirmation, the forwarding status of raw materials is checked | PM and MO check the status. |
| Set floor wise target | 2.3 | After receiving the raw materials, the production target for each floor is set | PM sets the production target. |

Use case diagram:



2.1: VIEW CONFIRMATION

|  |  |
| --- | --- |
| Use case name: | View confirmation |
| Use case id | 2.1 |
| Priority | High |
| Primary system actor | Raw material storage subsystem(RMS) |
| External receiver actor | Main office (MO),production manager(PM) |
| Description | Showing confirmation of sending raw materials |
| Trigger by | RMS |

Table: Use Case Narrative for view confirmation

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: submit a confirmation to system | Step 2: forward the confirmation to MO and PM |
|  | Step 3: send a notification to MO and PM |

Documentation of the events of the view confirmation use case:

Conclusion: concludes when a notification is sent to MO and PM

Post condition: PM and MO will confirm of receiving raw materials.

Implementation issues: system will notify MO and PM automatically.

2.2: check forwarding status

|  |  |
| --- | --- |
| Use case name: | Check forwarding status |
| Use case id | 2.2 |
| Priority | medium |
| Primary system actor | Production manager(PM) |
| Description | Showing the current status of raw material forwarding(whether it has reached factory, the quantity, condition of raw materials) |
| Trigger by | PM |

Table: Use Case Narrative for check forwarding status

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: PM requests to see the current status of raw materials | Step 2: system provides an answer whether it has reached factory of is in the way |

Documentation of the events of the check forwarding status use case:

Conclusion: concludes when a notification is sent to MO and PM

Post condition: PM and MO will confirm of receiving raw materials.

Implementation issues: system will notify MO and PM automatically.

2.3: SET FLOOR WISE TARGET

|  |  |
| --- | --- |
| Use case name: | Set floor wise target |
| Use case id | 2.3 |
| Priority | medium |
| Primary system actor | Production manager(PM) |
| Description | Setting a production target for each floor |
| Trigger by | PM |

Table: Use Case Narrative for set floor wise target

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: PM requests to see the current status of raw materials | Step 2: system provides detail about raw material. |
| Step 2: PM assigns production target for each floor | Step 3: It is recorded in system. |

Documentation of the events of the set floor wise target use case:

Conclusion: concludes when an entry is made.

Pre-condition: PM will receive confirmation of raw materials arrival.

Post-condition: a floor target will be set for each floor.

Implementation issues: system will provide detail about raw materials.

3. Production stage tracking:

This subsystem is the heart of our proposed system. In a garments industry, till now production stage tracking is all in all a manual process. Office ask about the production update to section supervisor, section supervisor then check the situation ( which is very tough as there is various sections in a factory like cutting, sewing, packing so certainly no process of real time tracking) and then inform the office. Our subsystem automates the whole tracking system and makes it easier to check the production status.

Actors for this subsystem:

1. Production manager

2. Section supervisor

3. Production workers

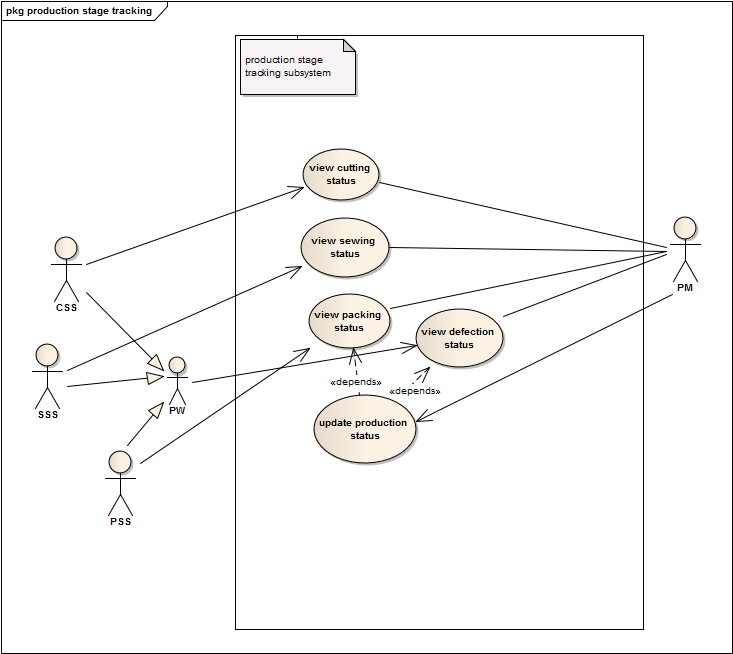
Actor’s glossary:

|  |  |  |
| --- | --- | --- |
| Actor name | Short key | Activity scope |
| Production manager | PM | updates the status of production, sends confirmation to MO, forwards the products to DM. |
| Section supervisor | SS | updates the status of production in his section, checks for defected products, sends report to PM |
| production workers | PW | inputs the finished and defected production status(abstract actor) |

Use case Glossary:

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Use case id | Description | Participant actors and roles |
| view cutting status | 3.1 | Checking and updating cutting section product status | cutting section SS updates , PM checks it |
| view sewing status | 3.2 | Checking and updating sewing section product status | sewing section SS updates , PM checks it |
| view packing status | 3.3 | Checking and updating packing section product status | packing section SS updates , PM checks it |
| view defection status | 3.4 | Checking and updating defected product status | CSS,SSS,PSS inputs, PM checks |
| Update production completion status | 3.5 | Updating total production status of the order | PM updates |

Use case diagram for Production stage tracking:



USE CASE NARRATIVE FOR VIEW CUTTING STATUS USE CASE:

|  |  |
| --- | --- |
| Use case name: | View cutting status |
| Use case id | 3.1 |
| Priority | Normal |
| Primary system actor | Cutting Section Supervisor(CSS) |
| External receiver actor | Production manager(PM) |
| Description | Checking and updating cutting section product status |
| Trigger by | CSS |

Table: Use Case Narrative for view cutting status target

Typical Course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: CSS requests to input report | Step 2: show an UI to update |
| Step 3: CSS inputs a status report | Step 4: delete previous report and store the new input |
| Step 5: PM checks for status | Step 6: show stored status |

Alternate course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 5: report a problem | Step 6: sends notification to production manager |

Documentation of the events of the view cutting status target use case:

Conclusion: concludes when all entries are made.

Post-condition: update will be stored in system

Implementation issues: system will provide a UI to make changes.

USE CASE NARRATIVE FOR VIEW SEWING STATUS USE CASE:

|  |  |
| --- | --- |
| Use case name: | View sewing status |
| Use case id | 3.2 |
| Priority | Normal |
| Primary system actor | Sewing Section Supervisor(SSS) |
| External receiver actor | Production manager(PM) |
| Description | Checking and updating sewing section product status |
| Trigger by | SSS |

Table: Use Case Narrative for view cutting status target

Typical Course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: SSS requests to input report | Step 2: show an UI to update |
| Step 3: SSS inputs a status report | Step 4: delete previous report and store the new input |
| Step 5: PM check for status | Step 6: show stored status |

Alternate course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 5: SSS report a problem | Step 6: sends notification to production manager |

Documentation of the events of the view cutting status target use case:

Conclusion: concludes when all entries are made.

Post-condition: update will be stored in system

Implementation issues: system will provide a UI to make changes.

USE CASE NARRATIVE FOR VIEW PACKING STATUS USE CASE:

|  |  |
| --- | --- |
| Use case name: | View packing status |
| Use case id | 3.3 |
| Priority | Normal |
| Primary system actor | packing Section Supervisor(PSS) |
| External receiver actor | Production manager(PM) |
| Description | Checking and updating packing section product status |
| Trigger by | PSS |

Table: Use Case Narrative for view packing status target

Typical Course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: PSS requests to input report | Step 2: show an UI to update |
| Step 3: PSS inputs a status report | Step 4: delete previous report and store the new input |
| Step 5: PM check for status | Step 6: show stored status |

Alternate course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 5: PSS report a problem | Step 6: sends notification to production manager |

Documentation of the events of the view packing status target use case:

Conclusion: concludes when all entries are made.

Post-condition: update will be stored in system

Implementation issues: system will provide a UI to make changes.

3.4 View Defection Status:

|  |  |
| --- | --- |
| Use case name: | View defection status |
| Use case id | 3.4 |
| Priority | Low |
| Primary system actor | CSS, SSS, PSS |
| External receiver actor | Production manager(PM) |
| Description | Checking and updating defected product status |
| Trigger by | CSS,SSS or PSS |

Typical Course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1:PSS/CSS/SSS requests to input a defection status | Step 2: show a form for entering defection status |
| Step 3: PSS/CSS/SSS inputs a defection status | Step 4: increase defection count by one and store into defection report center |
| Step 5: PM check defection status | Step 6: show defection percentage and defection report |

Documentation of the events of the view packing status target use case:

Conclusion: concludes when all entries are made.

Post-condition: update will be stored in system

Implementation issues: system will provide a UI to make changes.

3.5 Update Production Status:

|  |  |
| --- | --- |
| Use case name: | Update production status |
| Use case id | 3.5 |
| Priority | High |
| Primary system actor | Production manager(PM) |
| External receiver actor | Main Office (MO) |
| Description | Updating total production status of the order |
| Trigger by | PM |

Typical Course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: PM requests to update status | Step 2: show update UI |
| Step 3: PM updates status | Step 4: store status |
| Step 5: MO check for status | Step 6: show stored status |
|  | Step 7: sends notification to Main Office |

Documentation of the events of the view packing status target use case:

Conclusion: concludes when all entry is made.

Pre-condition: view packing status use case gives no error report.

Post-condition: update will be stored in system

Implementation issues: system will provide a UI to make changes.

1. DELIVERY OF PRODUCTS:

This subsystem deals with the delivery of products after completion of the production. The Delivery Manager informs the client that their goods are ready for delivery and sets the shipment date and process. After the products are sent for delivery, the Database Administrator updates the stock information but the delivery report option remains unchecked. When clients confirm that they have got all products alright, only then it is updated and the procedure is finished.

Actors for this subsystem:

• Delivery manager

• Database administrator

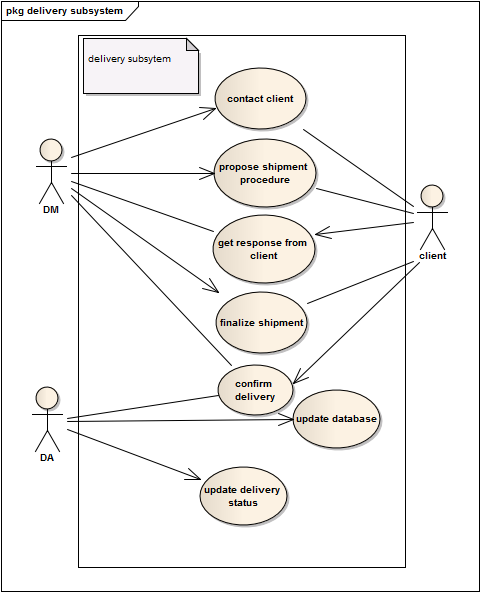
• Client

Actors Glossary:

|  |  |  |
| --- | --- | --- |
| ACTOR | SHORT KEY | ACTIVITY SCOPE |
| Delivery manager | DM | Informs clients about delivery and ensures the total process of delivery |
| Database Administrator | DA | Updates the Stock information in database |
| Client | CL | Places order and receives good after production |

Use case glossary:

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Use case id | Description | Participant actors and roles |
| Contact client | 4.1 | Contact is made to inform that products are ready for delivery | DM informs CL |
| propose shipment procedure | 4.2 | A suitable date, place and way of delivery is proposed and clients are informed | DM finds a suitable date, place, way for delivery and informs client |
| Get response from client | 4.3 | Client responds whether the proposal is okay or not. | Client confirms DM or demands any change |
| finalize shipment | 4.4 | If clients agree on the proposed date, delivery manager finalizes that and take necessary steps for delivery. Otherwise, proposes another suitable date, place, and way. | DA finalizes delivery process or repeats use case 'propose shipment procedure' |
| Update Database | 4.5 | Stock information is updated | DA updates stock information |
| Confirm delivery | 4.6 | Ensuring that products have been received soundly. | Clients confirm that they received products |
| Update delivery report | 4.7 | After getting confirmation, the process is ended by updating delivery report. | DA updates delivery status |



4.1: contact client

|  |  |
| --- | --- |
| Use case name: | Contact Client |
| Use case id: | 4.1 |
| Priority | High |
| Primary System actor | Delivery Manager(DM) |
| External receiver actor | Client |
| Description | Clients are informed that products are ready for delivery |
| Trigger by | DM |

Table: Use Case Narrative for contact client

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: get client info | Step 2: Show client details |
| Step 3: Inform client about delivery |  |

Documentation of the events of the user demand use case:

Conclusion: Concludes when client is informed that products are ready for delivery

Post condition: Clients get ready to receive the delivery

4.2: Propose Shipment procedure

|  |  |
| --- | --- |
| Use case name: | Propose Shipment procedure |
| Use case id: | 4.2 |
| Priority | High |
| Primary System actor | Delivery Manager(DM) |
| External receiver actor | Client |
| Description | Delivery Manager finds a suitable date for delivery and knowing the place of delivery finds a suitable way and then informs client about that. |
| Trigger by | DM |

Table: Use Case Narrative for 'propose shipment procedure'

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: Search for available dates | Step 2: Show available dates |
| Step 3: Select date | None |
| Step 4: Search possible paths to send delivery | Step 5: Show all possible paths to send delivery |
| Step 6: Select Path of delivery | None |
| Step 7: Inform clients about selected date and path |  |

Documentation of the events of the user demand use case:

Conclusion: concludes when client is informed about shipment procedure

Post condition: Clients consult about proposed procedure and let know DM whether it is okay or not.

4.3: Get response from client

|  |  |
| --- | --- |
| Use case name: | Get response from client |
| Use case id: | 4.3 |
| Priority | High |
| Primary System actor | Client |
| External receiver actor | Delivery Manager(DM) |
| Description | Clients confirms DM whether the proposed shipment procedure is okay or not. If it is not okay, then DM proposes another procedure. |
| Trigger by | Client |

Table: Use Case Narrative for 'Get response from client'

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: Client requests to input response | Step 2:show a form to input client’s response |
| Step 3: client saves his response | Step 4:system makes an entry |
|  | Step 5: system notifies DM |

Documentation of the events of the user demand use case:

Conclusion: concludes when DM gets informed about client’s opinion about shipment procedure

Post condition: DM finalizes the shipment procedure or proposes another one

4.4: Finalize Shipment

|  |  |
| --- | --- |
| Use case name: | Finalize Shipment |
| Use case id: | 4.4 |
| Priority | High |
| Primary System actor | Delivery Manager(DM) |
| External receiver actor | Client |
| Description | If clients agree on the proposed date, delivery manager finalizes that and take necessary steps for delivery. Otherwise, proposes another suitable date, place and way. |
| Trigger by | DM |

Table: Use Case Narrative for 'Finalize shipment'

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: DM checks Clients opinion | Step 2: Show clients opinion |
| Step 3: if ‘alright’, finalizes or goes back to 4.2 |  |

Documentation of the events of the user demand use case:

Conclusion: concludes when DM finalizes shipment procedure

Post condition: Products are sent for delivery according to finalized procedure.

4.5: Update Database

|  |  |
| --- | --- |
| Use case name: | Update Database |
| Use case id: | 4.5 |
| Priority | medium |
| Primary System actor | Database Administrator(DA) |
| Description | Once the products are sent for delivery, database is updated by subtracting those product quantities from stock information. |
| Trigger by | DA |

Table: Use Case Narrative for 'Update database'

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: DA updates database | Step 2: Information is updated |

Documentation of the events of the user demand use case:

Conclusion: concludes when stock information is updated

Post condition: Delivery report option remains still unchecked

4.6: Confirm delivery

|  |  |
| --- | --- |
| Use case name: | Confirm delivery |
| Use case id: | 4.6 |
| Priority | Medium |
| Primary System actor | Client, DM |
| External receiver actor | DM,DA |
| Description | Clients confirm DM that they received products and DM informs DA about it. |
| Trigger by | Client |

Table: Use Case Narrative for 'propose shipment procedure'

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: Client requests to record confirmation | Step 2:System shows a UI to record confirmation |
|  | Step 3: system notifies DM and DA |

Documentation of the events of the user demand use case:

Conclusion: concludes when Database Administrator gets informed about delivery status

Post condition: DA updates database

4.7: Update delivery status

|  |  |
| --- | --- |
| Use case name: | Update delivery status |
| Use case id: | 4.7 |
| Priority | Medium |
| Primary System actor | DA |
| Description | After getting confirmation from client, DA updates database and the sub-system is ended by updating delivery report as checked one. |
| Trigger by | DA |

Table: Use Case Narrative for ‘Update delivery status ’

Typical course of events:

|  |  |
| --- | --- |
| Actor action | System response |
| Step 1: DA updates Database | Step 2: database is updated |

Documentation of the events of the user demand use case:

Conclusion: concludes when Database is updated about delivery report

Post condition: The delivery details are kept as record in database for future calculation of Year-end report.