**Architectural Drivers for Willy’s Widget Company’s Spartan**

**Team 3 (Spartan)**

**REVISION HISTORY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S. No. | Revision Description | Modified  By | Modification Date | Comments |
| 1. | Initial draft created | Vijay R | 2014/05/12 |  |
| 2. | Use cases added  Quality Attribute Security added  Assumptions regarding Login credentials removed | Vijay R | 2014/05/19 |  |
| 3. | Refinement of Quality Attributes | Tommy Park | 2014/5/21 |  |
| 4. | Development Process added | Sun Shin | 2014/5/24 |  |
| 5 | Refinement of Quality Attributes | Tommy Park | 2014/5/24 |  |
| 6 | Modification of use cases from 1 to 4 | Gina Du | 2014/5/24 |  |
| 7 | Use cases from 5 to 12 | Jugwan Eom | 2014/5/25 |  |
| 8 | Team work  Use cases and functional requirements updated | Tommy Park | 2014/5/25 |  |
| 9 | Refinement of detail Use Cases from 1 to 4 based on team work | Gina Du | 2014/5/25 |  |
| 10 | Update Process according to Updated Use cases | Sun Shin | 2014/5/25 |  |
| 11 | Refinement of detail Use Cases from 1 to 4 based on team work | Jugwan Eom | 2014/5/25 |  |
| 12 | Team work  Quality attributes, constraints and assumptions updated. | Tommy Park | 2014/5/26 |  |
| 13 | Reflect the mentor’s comments | Gina Du  Jugwan Eom  Tommy Park | 2014/5/29 |  |
| 14 | Scope out added  Updated font  Added footer  Updated Assumptions | Vijay | 2014/05/29 |  |
| 15 | Teamwork  Fixed for initial presentation | All | 2014/5/29 |  |

Contents

Contents 3

1. Use Cases (High-Level …) 4

2. Functional Requirements 10

3. Constraints 12

3.1 Business constraints 12

3.2 Technical constraints 12

4. Quality Attributes 14

4.1 Selection of Quality Attributes 14

4.2 Quality Attribute Scenarios 16

4.2.1 Usability 16

4.2.2 Reliability 17

4.2.3 Scalability 17

4.2.4 Availability 17

4.2.5 Performance 18

5. Assumptions 19

6. Scope out 20

1. Use Cases (High-Level …)

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | Use Case Name | **Description** | **Actor** |
| UC01 | Make Order | User placing order to warehouse | Customer |
| UC02 | Monitor order status | Supervisor can monitor order processing status | Warehouse Supervisor |
| UC03 | Add inventory | Supervisor can add inventory items | Warehouse Supervisor |
| UC04 | Monitor inventory status | Supervisor can monitor inventory status | Warehouse  Supervisor |
| UC05 | Monitor robot status | Supervisor can monitor robot status | Warehouse Supervisor |
| UC06 | Process Order | Warehouse can handle order by interacting with warehouse H/W system and Warehouse system can update warehouse status | System |
| UC07 | Packing and Shipping Order | Worker packs and ships order to the customer | Shipping Workers |

|  |  |
| --- | --- |
| Use Case Title | Make an order |
| Use Case ID | UC\_001 |
| Primary actor | Customer |
| General Use Case description | User placing order to warehouse |
| Entities involved | Customer, Order System, mobile device (or laptop), Warehouse System |
| Preconditions | Customer has an access authority to order system.  Customer has been already connected in order system |
| Primary use case flow of events | 1. Customer selects widget type from Customer Application 2. Customer enters quantity of widgets, address and phone number 3. Customer presses a complete button to submit the order. 4. Customer Application sends the order to Warehouse System. 5. Customer Application displays the status of order to customer. 6. Customer connects the Customer Application to checks tracking of order status. |
| Post conditions | All order items have been fully written.  Warehouse system takes an order and updated.  Order system displays status of the order for customers. |
| Alternative use case | 4-1. If you do not complete at least one of them, Customer Application displays a warning message to customer.  4-2. Customer Application displays error messages like invalid input, network problems, etc. |
| Use Case Title | Make an order |

|  |  |
| --- | --- |
| Use Case Title | Monitor order status |
| Use Case ID | UC\_002 |
| Primary actor | Warehouse Supervisor |
| General Use Case description | Supervisor can monitor order processing status. |
| Entities involved | Order System, Warehouse System, Supervisor |
| Preconditions | All ordering items are completed.  Supervisor has an access authority to monitoring and management system.  Supervisor has been already connected in monitoring and management system.  Supervisor is able to add widgets anytime if he wants, does not fail to add widgets. |
| Main success scenario | 1. Warehouse Supervisor System displays the status of Warehouse System with “In-Process” message with processing order’s items.  2. Supervisor checks the items to make up an order as pressing a given specific button from Warehouse Supervisor System if backordered order is exist.  3. Supervisor add widgets at stations to process backordered order. |
| Postconditions | Warehouse system displays status of all orders.  Customer Application system is able to update the status of order when it is changed. |
| Alternative use case | 1-1. Warehouse Supervisor System displays a pending message when robot carrier is busy, so order is waiting for resources.  1-2. Warehouse Supervisor System displays a backordered message not enough inventory to complete order.  1-3. Warehouse Supervisor System displays a complete message when order has been filled.  2-1. Warehouse Supervisor System makes dimmed button if backordered order doesn’t exist. |

|  |  |
| --- | --- |
| Use Case Title | Add inventory |
| Use Case ID | UC\_003 |
| Primary actor | Warehouse Supervisor |
| General Use Case description | Supervisor can add inventory items |
| Entities involved | Supervisor, Warehouse System, Supervisor Application, Customer application |
| Preconditions | Supervisor has an access authority to create inventory entry to Application System.  Supervisor has been already connected in Supervisor Application.  There is out of stock in Warehouse System. |
| Main success scenario | 1. Supervisor creates new inventory entry for inventory stations.  2. Supervisor enters the quantity of each item, item description, cost and station number.  3. Warehouse Supervisor Application displays a success message of adding new inventory to warehouse system.  4. Warehouse Supervisor Application sends update type of widgets for added new inventory to Customer Application. |
| Postconditions | New inventory entries are fulfilled.  Added items are moved and stacked at stations.  Update current count and widgets at the stations at Warehouse System.  Robot gets items that are added from inventory and management system.  Customer is able to select added inventory from Customer Application. |
| Alternative use case | 1-1. If you do not complete at least one of them, display warning message to supervisor.  2-1. If Warehouse system cannot create new inventory entry, Supervisor Application has to handle these situations. (ex. Display warning message)  3-1. Warehouse System cannot add widgets as like exceed maximum number of widgets, Supervisor Application handles it. (ex. display warning message to supervisor and go back to previous screen, or delete menu)  3-2. Disconnected Warehouse System add some widgets, supervisor is able to know it. (ex. Display a warning message for supervisor) |

|  |  |
| --- | --- |
| Use Case Title | Monitor inventory status |
| Use Case ID | UC\_04 |
| Primary actor | Warehouse Supervisor |
| General Use Case description | Supervisor can monitor inventory status |
| Entities involved | Warehouse Supervisor, Warehouse System, Supervisor application |
| Preconditions | Supervisor has been already connected in Supervisor Application |
| Main success scenario | 1. Supervisor enter monitoring screen  2. Supervisor sees the available inventory at each station  3. If items are loaded onto the robot, inventory will be automatically debited and supervisor will be notified via monitoring screen |
| Postconditions |  |
| Alternative use case |  |

|  |  |
| --- | --- |
| Use Case Title | Monitor robot status |
| Use Case ID | UC\_005 |
| Primary actor | Warehouse Supervisor |
| General Use Case description | Supervisor can monitor robot status |
| Entities involved | Warehouse supervisor, robot |
| Preconditions | Supervisor has an access authority to Supervisor Application.  Supervisor has been already connected in monitoring system |
| Main success scenario | 1. The Supervisor Application shall display the robot information as following   * location of the robot (station number) * Stations visited by the robot (for single order) * next station to be visited. * Status (in-process, idle, error) * In-Process: robot is filling an order * Idle: no orders to fill * Error: problem – include a description of the problem |
| Postconditions | - |
| Alternative use case | - |

|  |  |
| --- | --- |
| Use Case Title | Process Order |
| Use Case ID | UC\_006 |
| Primary actor | Warehouse System |
| General Use Case description | System processes and completes customer's orders |
| Entities involved | System, Robot |
| Preconditions | Customers place orders (UC\_001) |
| Main success scenario | 1. System gets an order  2. System changes the order status to 'In-processes'  3. System decides next visited inventory station according to the order and inventory status  4. System moves robot to the inventory stations where order items will be loaded  5. System updates robot status  6. At each station, inventory worker loads a item on bot  7. System updates inventory status (debit) and robot status  \* Repeat Steps 6-7 until all items at the station are loaded  8. Worker presses a button at the station to indicate loading is complete  \* Repeat Steps 3-8 until all ordered items are loaded on bot  9. System moves the robot to shipping area and updates robot status  10. Warehouse worker at shipping area unloads all items from bot  11. System updates robot status  12. Worker presses a button that indicates the order is unloaded and completed  13. System changes the order status to 'Complete'  \* Repeat steps 1-13 as long as there are orders to fill |
| Post conditions | All placed orders are completed |
| Alternative use case | 2-1. If the robot is not idle, system changes the order status to 'pending' and waits for the robot  2-2. If there is not enough inventory to complete the order, system changes the order status to 'Backordered' and using UC\_003  2-2-1 System processes next order  3-1. If robot could not moved to next station during predefined time, system handles the error case  8-1. If robot could not moved to next station during predefined time, system handles the error case |

|  |  |
| --- | --- |
| Use Case Title | Packing and Shipping Order |
| Use Case ID | UC\_07 |
| Primary actor | Shipping Workers |
| General Use Case description | Worker packs and ships order to the customer |
| Entities involved | Shipping Worker, Warehouse System |
| Preconditions | Processing order is complete (UC\_06) |
| Main success scenario | 1. Shipping worker packs all the items to a box and ships it to the customer |
| Post conditions | The order is shipped to the customer |
| Alternative use case |  |

1. Functional Requirements

|  |  |  |
| --- | --- | --- |
| **UCID** | **FRID** | **Functional Requirement** |
| UC01 |  | Make Order |
|  | FR01 | The Customer Application shall display the Type of the Widgets to the customer |
|  | FR02 | The Customer Application shall send the order to the Warehouse System |
|  | FR03 | The Warehouse System shall take the orders from the customers from the mobile device (laptop) and the order shall include the following   * Type of the Widget * Quantity of each Widget * Address of the customer * Phone number of the customer |
|  | FR04 | The Customer Application should display the status of order. |
|  | FR05 | The Customer Application can handle error cases (invalid input, network disconnection.) |
| UC02 |  | Monitor order status (Supervisor) |
|  | FR01 | The Supervisor Application shall display the Order status as following   * showing the items of an order * pending: waiting for resources (the robot carrier is busy) * in-processes: order is currently being filled * backordered: not enough inventory to complete order (all or nothing policy) * complete: the order has been filled (e g. unloaded) |
|  | FR02 | Supervisor Application should provide an interface to handle backordered order (e.g., showing shortage of types of widgets) |
| UC03 |  | Add inventory |
|  | FR01 | The Supervisor Application shall provide an interface to create new inventory entry for Inventory Stations (with the quantity of each item and item description, cost, station number) |
|  | FR02 | The Supervisor Application shall provide an interface to add the number of widgets in Inventory Stations. |
| UC04 |  | Monitor Inventory status |
|  | FR01 | The Supervisor Application shall display the Inventory and also the Inventory available at each station   * the quantity of each item and item description, cost, station number |
| UC05 |  | Monitor robot status |
|  | FR01 | The Supervisor Application shall display the robot information as following   * location of the robot (station number) * Stations visited by the robot (for single order) * Next station to be visited. * Status (in-process, idle, error) * In-Process: robot is filling an order * Idle: no orders to fill * Error: problem – include a description of the problem |
| UC06 |  | Process Order |
|  | FR01 | The Warehouse system should maintain customer orders with Queue (FIFO) |
|  | FR02 | The Warehouse system should maintain backordered orders with another Queue. |
|  | FR03 | The Warehouse contains a robot. The Warehouse System shall move the robot to the next station till all the items of the order are loaded. |
|  | FR04 | (The Warehouse worker will load the items on the robot at an inventory station and press the button.) |
|  | FR05 | When Inventory worker press the button, the warehouse system debits corresponding widgets from entries in Inventory. |
|  | FR06 | The system shall move the robot to the shipping station after the order is filled and the items will be unloaded by the Warehouse worker. . |
|  | FR07 | (The Shipping worker will unload the items on the robot and press the button.) |
|  | FR08 | When the shipping worker presses the button, the warehouse system updates order status and robot status |
|  | FR09 | When an error happens, the supervisor can order recovery action. (Controlling robots manually, Displaying situation to supervisor applications… , ….) |
| UC07 |  | Packing and Shipping Order |

1. Constraints

3.1 Business constraints

| **ID** | **Name** | **Description** |
| --- | --- | --- |
| BC1 | Time to market | * Deadline: 27th of June * Only 5 weeks and a lot of obstacle events (weekly exam & activities) |
| BC2 | Lack of Resource | * There is only one warehouse kit * We are short of 1 human resource. |
| BC3 | Programming Expertise | * Most of team members are not familiar with Java. * No one has experience of arduino programming |
| BC4 | Realization | * Software architecture and reference implementation can be applied to other companies which have similar problems. |
| BC5 | Ambiguity of Customer | * Customers are not actual customers, so that their requirements are ambiguous. |
| * Need to develop simple working prototype and expand it to meet the time constraints. * Need to develop warehouse emulation component * To overcome the lack of expertise, need to start arduino and complicated Java part in early stage * Must not use unrealistic assumption * Supporting Future Needs of Multiple Robots and multiple inventory stations. * We need to find hidden requirements from customers. | | |

3.2 Technical constraints

| **ID** | **Name** | **Description** |
| --- | --- | --- |
| TC1 | Arduino | * Need to use arduino 1.0.4 SDK for developing software that sits in robot and warehouse |
| TC2 | OS | * Applications should be reside in a Window PC |
| TC3 | Programming Language | * Need to use Java language for customer application and monitoring application |
| TC4 | Database | * MySQL shall be used for warehouse DB. * Need to support IP-based connection. |
| TC5 | Network | * Communication between software entities can be made over IP-based network |

1. Quality Attributes

4.1 Selection of Quality Attributes

Since we have only 5 weeks to rollout the first implementation, we need to focus on one or two quality attributes that can be representative of this system. We set three metrics to prioritize quality attributes, (P)ossibility, (T)ime-consuming, and (I)mpact to customer. We only select quality attributes that has high possibility and impact and can be done within 5 weeks. Others are remained for the future consideration.

| **Quality Attributes** | **Description** | **P** | **T** | **I** | **Priority** |
| --- | --- | --- | --- | --- | --- |
| Usability | Easy to use for the customer, Warehouse Supervisor, Warehouse Inventory Manager   * Customer can submit order and check the order status anywhere * Submitting order shall be less than 3 steps excluding authentication steps. * Supervisor can see all the status of warehouse at a glance | High | Middle | High | 1 |
| Scalability | The system is allowed to add components and its performance improves proportionally after adding components   * Robot and trace line can be added * Inventory Station can be added * Type of widgets can be added * Server capacity can be increased | Middle | Low | Middle | 3 |
| Performance | The system should be considerably fast adding the inventory, placing the orders, updating the order status, the robot status and the overall status.   * Adding inventory within 1 sec * Update status of order, robot, and inventory within 1 second * Taking order within 1 sec * Processing an order that has maximum capacity of the robot (loading and shipping) within OO min | High | High | Low | 5 |
| Availability | The system should not hang or get unusual in any situation, so that customers and warehouse supervisor should be able to use the system at any point of time.   * Warehouse Server should not be down if it receives abnormal request * Warehouse Server should not be down if it receives more requests than its capacity: (it allows 5 users to connect the server simultaneously) | High | Middle | Middle | 4 |
| Reliability | The system should guarantee that order transaction to be completed or backordered once it starts. | Middle | Middle | High | 2 |
| Extensibility | The system should be extended to following environments   * Customers submit order with mobile devices | Middle | Middle | Middle | 6 |
| Security | The system should not allow the customer or Warehouse supervisor without any valid login credentials | Low | High | Low | Not considered |

4.2 Quality Attribute Scenarios

4.2.1 Usability

Usability is the ease-of-use of the software product. Detailed usability scenario for the

system is described in this section.

|  |  |
| --- | --- |
| Id | QA01 |
| Quality Attribute | **Usability (**Easy to use for the customer, Warehouse Supervisor, Warehouse Inventory Manager) |
| Stimulus: | Launch application’s UI and Using system functions (taking orders and monitoring order/robot/warehouse status) with these interfaces. |
| Source: | Customer or Warehouse Supervisor |
| Environment: | Runtime |
| Response: | Display the Order Status to customer  Display the overall status and robot status to Warehouse Manager/Supervisor |
| Response measure: | Comfortable and easily understandable display of Inventory, Status  Number of Steps of Order Submission  Number of Steps for Monitoring  Undo/Reset/Modify Functionality  Progress Indicator  Operation cancellation/modification allowed  Ease-of-navigation (Back and Forward) |

4.2.2 Reliability

|  |  |
| --- | --- |
| Id | QA02 |
| Quality Attribute | **Reliability** (The system should guarantee that order transaction to be completed or backordered once it starts.) |
| Stimulus: | Error happens in the system and order is not being processed |
| Source: | System module (robot, inventory worker) |
| Environment: | Normal |
| Response: | Detect error. Resolve it or rollback the transaction. |
| Response measure: | Request Complete Ratio > 99% |

4.2.3 Scalability

|  |  |
| --- | --- |
| Id | QA03 |
| Quality Attribute | **Scalability** The system is allowed to add components and its performance improves proportionally after adding components |
| Stimulus: | 1) Add robot, 2) add warehouse, |
| Source: | Warehouse Owner |
| Environment: | Prepared for deployment of added component (robots, stations, widgets) |
| Response: | Processing order |
| Response measure: | 1) Throughput [increased number of order processed / added robot] > TBD  2) Throughput [increased number of order processed / added warehouse] > TBD |

4.2.4 Availability

|  |  |
| --- | --- |
| Id | QA04 |
| Quality Attribute | **Availability** The system should not hang or get unusual in any situation, so that customers and warehouse supervisor should be able to use the system at any point of time. |
| Stimulus: | 1) Warehouse server meets an abnormal request, 2) Warehouse server meets heavier request than its capability |
| Source: | Customer |
| Environment: | Normal Runtime |
| Response: | The server replies with the corresponding error message and continue to work |
| Response measure: | The server is not crashed while processing a hundred and more requests |

4.2.5 Performance

|  |  |
| --- | --- |
| Id | QA05 |
| Quality Attribute | **Performance** The system should be considerably fast adding the inventory, placing the orders, updating the order status, the robot status and the overall status. |
| Stimulus: | 1) Add an inventory, 2) Take an order 3) Update the order status and displayed 4) Process an order |
| Source: | 1)2)3) Supervisor 4) System |
| Environment: | Normal Runtime |
| Response: | Operation completed within expected time |
| Response measure: | 1) Response time of adding inventory (< 1 sec)  2) Response time of taking an order (<1 sec)  3) Response time of updating any status and being displayed (<1 sec)  4) Response time of processing an order (< TBD min) |

1. Assumptions
2. Orders will be processed sequentially (one after the other. Not more than one order can be processed)
3. Single installer is not required
4. No two stations will have the same type of widget. That is, the same type of widget will be located only at a single station.
5. Scope out
   * + 1. Login for Customer and Supervisor
       2. Cancellation of order by the customer