Customer Priority Maintenance Screen	
Functional Area:	Scheduling/Production
Proposed Release:	R27
Business Process:	Prioritising orders
Process Objective:	Ability to maintain customer priority information in Halo
Product Backlog Item (PBI):	10398
Estimated cost of development:	**£2,000**
Estimated cost of non-development to business:	Unknown
Estimated Business Value:	Unknown

1. Business Requirements

1. Ability to maintain customer priority information in Halo

2. Acceptance Criteria

2.1. Functional Acceptance Criteria

□ **2.1.1 – Customer Priority Maintenance**

ℹ The SAP A1 implementation of Halo allows for customer priority to be placed on production orders. When this functionality was first introduced, customer priorities had to be set up and maintained directly in the SQL tables. The business would now like to be able to maintain this information from inside the Halo application.

□ A new maintenance screen shall be introduced under the **Production Reporting >> Maintenance** menu called ‘Manage Customer Priorities’

□ The new maintenance screen shall provide basic CRUD operations for the Aurora CUSTOMER_PRIORITY table.

□ SITE_CODE shall not form part of the maintenance but shall instead be taken from the site code that Halo is currently executing against.

□ CUSTOMER_CODE shall be a dropdown list of valid customer codes retrieved through the implemented customer service (e.g. SAP A1, SAP B1 etc.)

□ PRIORITY shall be a number between 1 and 9999 where 1 is the highest priority and 9999 is the lowest possible priority.

□ It shall not be possible to have the same priority for 2 customers.

□ A unique key shall be added to the CUSTOMER_PRIORITY table on the PRIORITY field.

□ Appropriate validation shall be performed in the UI to prevent hitting the Aurora database with invalid data (e.g. data that violates key constraints, will be truncated or isn’t of the correct data type)

□ Data shall be loaded in a background thread in order to keep the UI responsive and shall load automatically upon opening the menu.

□ A refresh button shall be made available on the toolbar in order to prevent having the user close the form and reopen it to see the latest data.

	□ Built in menu level security shall be considered sufficient to allow access to this maintenance screen. If a user has access to the menu then they have the ability to create, read, update and delete records from the database.
3.0	User Interface Standards
3.0.1	□ Any new UI shall be introduced with a look and feel consistent with that of the area of Halo in which it is introduced.
3.1	Security / Preferences
3.1.1	□ No additional/new security or preferences shall be required to perform any of the other actions outlined in this specification.
3.2	Performance
3.2.1	□ Existing Halo performance shall be unaffected by the changes outlined in this specification.
3.3	Testing
3.3.1	□ An indication that development has been completed successfully shall be observed by successfully testing each test case in the test plan (seen in Appendix 1).

Technical Environment where the PBI will be deployed	
Description	
ERP / Systems	SAP / Aurora
Supported Database(S)	SQL2008 R2 SP2 and SQL 2012 SP1
Planned Release	Halo 27
Other Information	

Document Sign Off			
Name	Role	Date	Signature
Dominic Musgrave	Delivery		
James Pritchett	Development		

Document Sign Off
"*I agree that this set of requirements represents our best understanding of the requirements for the next portion of this project and that the solution described will meet our needs as we understand them today. I agree to make future changes in this baseline through the project's defined change process. I realise that changes might require us to renegotiate cost, resource, and schedule commitments.*"

Document History			
Version	Description	Date	Author
1.0	Initial Draft	09/06/16	Scott Davidson

Appendix 1 – Test Plan

It should be possible to run through the following tests from start to finish without any errors.

Test 1 – Customer Priority Maintenance / Create, Save and Validation

1. Navigate to **Production Reporting >> Maintenance >> Manage Customer Priorities**
2. **Observe** that data is loaded in the background and that you can access other Halo menus while data is loading.
3. **Create** a new record by pressing the add button.
4. Select a customer from the dropdown list and enter a priority **between 1 and 9999**
5. **Ensure** that you cannot enter a negative priority, a priority of 0 or a priority higher than 9999
6. **Ensure** that you cannot enter a decimal priority or a priority that contains lettters.
7. **Ensure** that you cannot enter your own customer code and that you must select a value from the dropdown list.
8. **Save your changes** and ensure that data is persisted and you do not receive any SQL errors.

Test 2 – Customer Priority Maintenance / Update and Delete

1. Navigate to **Production Reporting >> Maintenance >> Manage Customer Priorities**
2. Update an existing record by changing the priority to a different value.
3. **Save your changes** and ensure that data is persisted and you do not receive any SQL errors.
4. **Delete** the record you just saved and ensure you do not receive any SQL errors.

Test 3 – Customer Priority Maintenance / Duplicate Keys

1. Navigate to **Production Reporting >> Maintenance >> Manage Customer Priorities**
2. **Create** a new record by pressing the add button.
3. **Copy** the details of an existing record into the new record (same customer code and priority)
4. **Attempt** to save the record.
5. **Ensure** you are given a user friendly error message that you cannot save the record because a record already exists that is the same.
6. **Ensure** you do not receive a duplicate key violation error message from SQL.

Test 4 – Thread Handling

1. Navigate to **Production Reporting >> Maintenance >> Manage Customer Priorities**
2. While the form is loading quickly close it and wait a few minutes.
3. **Ensure** you do not receive any error messages due to the form being disposed before the data had finished loading.