|  |  |
| --- | --- |
| DENTALENS | |
| Last Edited on | 1/21/2019 | Test Strategy Document |
| Document Version | DRAFT| Version 0.3  Prepared By | Manmeet Chadha | |

**Template Control**

| Version | Date | Reviewer Name and Role | Approve by Name and Role |
| --- | --- | --- | --- |
| 0.2 | 12/13/2018 |  |  |
| 0.3 | 1/21/2019 |  |  |

Table of Contents

[1 Document Control 3](#_Toc532472455)

[1.1 Version History 3](#_Toc532472456)

[1.2 Review/Approval History 3](#_Toc532472457)

[1.3 Open Items 3](#_Toc532472458)

[2 Introduction 4](#_Toc532472459)

[2.1 Project Identification 4](#_Toc532472460)

[2.2 Project & Testing Summary 4](#_Toc532472461)

[3 Features To Be Tested 4](#_Toc532472462)

[4 Features OUT OF SCOPE from testing 5](#_Toc532472463)

[5 Approach – Testing Types 6](#_Toc532472464)

[5.1 Unit/Component and E2E Testing 6](#_Toc532472465)

[5.1.1 Database updates for dental claims 6](#_Toc532472466)

[5.1.2 Sending claims to DentaLens 6](#_Toc532472467)

[5.1.3 Completed claims in DentaLens 7](#_Toc532472468)

[5.1.4 Reading and Processing Dentalens Output Data 7](#_Toc532472469)

[5.1.5 Client EOB 8](#_Toc532472470)

[5.1.6 Historical Claims 9](#_Toc532472471)

[5.1.7 Error Handling 9](#_Toc532472472)

[5.2 Regression Testing 11](#_Toc532472473)

[5.3 Performance/Volume Testing 11](#_Toc532472474)

[6 Pass / Fail Criteria 11](#_Toc532472475)

[6.1 Approval Criteria 11](#_Toc532472476)

[7 Testing Controls & Procedures 11](#_Toc532472477)

[7.1 Test Deliverables 11](#_Toc532472478)

[7.2 Testing Tasks 11](#_Toc532472479)

[7.3 Key Roles and Responsibilities 11](#_Toc532472480)

[7.4 Resources 13](#_Toc532472481)

[7.5 Schedule 13](#_Toc532472482)

[7.6 Testing Success Criteria 13](#_Toc532472483)

[7.6.1 Entrance Criteria 13](#_Toc532472484)

[7.6.2 Exit Criteria 13](#_Toc532472485)

[7.7 Defect Management 13](#_Toc532472486)

[7.7.1 Defect Management Severity Definitions 14](#_Toc532472487)

[7.7.2 Defect Management of Testing Completion 14](#_Toc532472488)

[7.8 Risk Management 15](#_Toc532472489)

[7.8.1 Risks and Assumptions 15](#_Toc532472490)

[7.9 Progress Reporting 15](#_Toc532472491)

[8 Environmental Requirements 15](#_Toc532472492)

[8.1 Test Environment Provisioning Request Process 15](#_Toc532472493)

[9 References 16](#_Toc532472494)

[10 Glossary 16](#_Toc532472495)

# Document Control

## Version History

| Version No. | Date | Author | Change Description |
| --- | --- | --- | --- |
| 0.1 | 11/26/2018 | Manmeet Chadha | Initial Draft |
| 0.2 | 12/13/2018 | Manmeet Chadha |  |
| 0.3 | 1/21/2018 | Manmeet Chadha | Added the field ‘Mpi\_Factor\_Code’ to section 5.1.3 for verification of claims in the results\_output table.  Section 5.1.7 – Expected Behavior modified/added for scenario 3.3, 3.4 and 3.5 Error Handling scenarios. |

## Review/Approval History

| **Date** | **Name, Title** | **Role, Department** | **Version #** | **Document Responsibility** | **Notes/Attachments** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Open Items

| **ID** | **Description** | **Date entered** | **Status** | **Resolution** | **Resolution Date** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Introduction

## Project Identification

|  |  |  |
| --- | --- | --- |
| Project Name | Project Number | Date Created |
| DentaLens | eTime Code: I | 12/13/2018 |
| Project Sponsor | Project Owner | |
| IT Management | Meenu Talwar | |
| Program Manager | Project Manager | |
| Meenu Talwar | Mahender Singh | |
| Project Assessment | | |
|  | | |

## Project & Testing Summary

The purpose of this project is to implement a Payment Integrity product for client’s priced and non-priced dental claims and provide recommendations on the technical & clinical merits of the claim, at a line level, so that it can be paid properly and in accordance with the industry billing standards.

Features To Be Tested

The SQA team will validate the functional, end to end claim, prior to release.

Below are the features that will be verified as part of the ‘DentaLens’ testing:

* Mocking up creation of Dental claims for routing to DentaLens.
* Data fields added to Multiplan tables for dental claims.
* Sending claims to Dentalens system from Multiplan and staging tables’ updates.
* Mapping of attributes to fields in the DentaLens staging table.
* Verify completed claims in DentaLens tables
* Updates to output staging tables
* Returning claims to Multiplan and not updating the output of those claims
* Returning claims to Multiplan and updating the output after Multiplan has read the initial output.
* Returning claims to Multiplan and updating the output before Multiplan has read the initial output.
* Reading Dentalens output data. Below fields from the ‘results\_output’ table to be read by Multiplan:
* Result
* Factor\_Code
* Findings
* RYG\_Status
* NewCode
* Is\_Processed
* Correlation\_Id
* Mpi\_factor\_Code
* Verify claim and line reason/remark codes derivation after receiving DentaLens output.
* Verify claim and line flags derivation after receiving DentaLens output.
* Allowed/Savings calculation in UCW, for claims returned from DentaLens.
* Priced and Not Priced claims routing to DentaLens.

Features OUT OF SCOPE from testing

* 837D load process
* 837D claims validation
* 837D extract to client
* Client EOB generation
* Claim Exclusions – Claims being flagged as Claim Type=X.
* DentaLens factor Algorithms
* Fields in source\_input table that are marked MVP Optional and not avlbl with MPI.
* Historical Claim file/data validation and stripping PHI will not be done.
* Dental Client Product Type Setup in UCW
* Client Options – Presenting information to clients based on options they are setup with
* Passing Multiplan derived values to DentaLens platform, for the purpose of DL Portal
* DentaLens portal integration
* Functionality as to how the product ‘xDentalReview’ should function.
* Historical Claims load to the DentaLens system.

CLIENTS to be used for both Functional and E2E testing :

| **Client** | **Alternate\_acct\_id** | **Acct\_Id** | **CDM Setup** | **Product Hierarchy in XN table** |
| --- | --- | --- | --- | --- |
| Client\_1 | 1129000000 | 1129010001  CCode- MARSP3 R1 | xDentalReview | xDentalReview (RO=0)  OR  xDentalReview (RO=2)  OR  xDentalReview (RO=8) |

Approach – Testing Types

## Unit/Component and E2E Testing

In the UCW DB, the below tables need to be setup for the clients sending dental claims, in the CLAIMS\_CONFIG schema:

1. Account\_Profile
2. Claim\_Exchange
3. External\_Network
4. Wkflow\_process\_reg (Product setup only)

### Database updates for dental claims

* Verify dental claim data populated in tables Dental\_Claim and Dental\_Claim\_Line
* ~~Verify if Claim\_Type=’D’ is populated for dental claims.~~
* Claims will be routed to Dentalens based on the External\_Network table setup. No check for Claim\_Type=D.

### Sending claims to DentaLens

* Verify claims and line data is mapped to table source\_input for dental claims.
* Verify the mapping of the below data:
  + Provider data
  + Patient data
  + Insured data
  + Claim procedure
  + Claim lines
  + Remarks
* Verify loading of claims with mandatory fields missing.
* Verify updates to field ‘is\_processed’, when claim is read by DL for processing.
  + Is\_processed=0 (Implies claim is ready for DentaLens processing)
  + Is\_processed=1 (Implies claim is in-process within DentaLens system)

(Now ‘is\_processed’ is not part of table source\_input. It is instead part of pre-staging table. What is the need in the pre-staging table because records from that table will anyways be deleted when moved to source\_input. Does the above implication still hold true?)

### Completed claims in DentaLens

* Verify the below operations are performed on the claims using **‘runAnalyticsForBatch’** in UCWDentaLensAnalyticsSvc OSB proxy.
  + Copy from prestaging to staging
    - update Dentalens.dbo.source\_input\_staging set is\_processed = 'Processing' where is\_processed is null.
    - Insert records into staging table source\_input.
    - delete from Dentalens.dbo.source\_input\_staging where is\_processed = 'Processing'
  + Invoke SQL Server procedure (**g\_00\_checkDataDumpCompletion)** 
    - Truncates data from staging table.
  + Delete from prestaging
* Verify updates to field ‘is\_processed’ in results\_output table, when DL has finished processing the claim.
* Verify updates to field ‘is\_processed’ in results\_output table, when Multiplan has read the required data from the table.
* The below fields will be verified for claims in the results\_output table:
  + Result (A – Accept, D- Deny, I – Invalid)
  + RYG\_Status (Red, Yellow, Green)
  + Factor\_Code
  + NewCode (Will be populated with the recommended procedure code)
  + Correlation\_ID
  + Findings
  + Mpi\_factor\_Code
* Verify updates to field ‘is\_processed’ when DL updates outcome of a claim/line after initial update
  + If Multiplan has already read the initial output and shared results with the client.
  + If Multiplan has not yet read the initial output.
* Verify the below values for ‘is\_processed’ field in table results\_output.
  + **is\_processed = 0** means record is ready for DL process for marking of factor codes and ryg\_status (Cannot be verified explicitly. Sonia confirmed.)
  + **is\_processed = 1** means DL process marked the results and now ready for MPI team to fetch it
  + **is\_processed = 2** means record is read and results fetched by MPI. MPI process will update the value from 1 to 2 after reading it.
  + **is\_processed = 3** means DL process re-marked the results after execution of day end analytics. Multi-plan team is not going to read this record again. – (HOW is this done? – Some Special process)
  + is\_processed=-1 means that some line results into an invalid scenario.
* Verify analytics information is successfully retrieved from table Dentalens.dbo.results\_output to the UCW datastore, using **‘RetrieveAnalytics’** operation on UCWDentaLensAnalyticsSvc OSB proxy.

### Reading and Processing Dentalens Output Data

* Verify updates to field ‘is\_processed’ when Multiplan has read the required data from the table.
* The DL fields ’mpi\_factor\_code’, ‘NewCode’,’Findings’,’ryg\_status’ , ‘result’ to be populated in Multiplan DB fields Claims\_Active.Dental\_Claim\_Line.Factor\_Code, Claims\_Active.Dental\_Claim\_Line.New\_Proc\_Code, Claims\_Active.Dental\_Claim\_Line.Findings, Claims\_Active.Dental\_Claim\_Line.Factors\_Color\_Cd, Claims\_Active.Dental\_Claim\_Line.result respectively.
* **Line Level** - Verify Multiplan sets Line flags with the below possible values:

1. No issues on the line

D – Deny Line

M – Downcode/ Modify Line

S – Suspect Line

I - Invalid Line

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Input from DentaLens** | | | **Line Level Calculations at Multiplan** | | |
| **Result** | **RYG\_Status** | **NewCode** | **Line Flag** | **Allowed** | **Savings** |
| A | All Green | N/A | A | Client Allowed  Client Billed/Covered | $0 |
| A | With minm one yellow | N/A | S | Client Allowed  Client Billed/Covered | $0 |
| D | N/A | NULL | D | $0 | Client Allowed  Client Billed/Covered |
| D | N/A | NOT NULL | M | $0 | Client Allowed  Client Billed/Covered |
| I | N/A | N/A | I | $0 | Client Allowed  Client Billed/Covered |
| Not A/D/I  Or  Any other Invalid data combination | | | Route Claim to Hold/Release process  (Results\_output.is\_processed=-1)  Configured group will be notified via email. System configuration.  Once data corrected and is\_processed indicator value set to 1, results will be reprocessed.  (Claim Correction can only be done by the DL team) | | |

* **Claim Level** - Verify Multiplan sets Claim level remark code based on the line level flags and amounts derived.

DN0 = Pay claim (*no savings, no issues*)

DL0 = Pay claim (*no savings, but may have other issues identified at the line level such as suspect*)

DL1 = Deny full claim

DL2 = Partial deny with no reason for re-repricing (*some savings and other combination of line flags but no occurrence of ‘M’ line*)

DL3 = Partial deny with reason for re-repricing (*some savings and other combination of line flags and occurrence of ‘M’ line*)

*\*DN0 can be mapped to DL0, if client doesn’t want the distinction*

*\*DL2 can be mapped to DL3, if client doesn’t want the distinction*

|  |  |  |
| --- | --- | --- |
| **Total line Savings** | **Line Flags** | **Claim Remark Code** |
| =$0 | All line flags = A ~~or NULL~~ | DN0 |
| =$0 | All line flags<> A ~~or NULL~~  Can have any line flag = S | DL0 |
| >$0 | All line flags only D or I | DL1 |
| >$0 | Not all line flags D or I  No line flag = M | DL2 |
| >$0 | At least one line flag = M | DL3 |

### Client EOB (Out Of Scope)

* Verify the Client EOB populated with the high level classification corresponding to factors on individual lines.
* Verify the Client EOB populated with the 50-char short description corresponding to the factors on the individual lines
* Verify high level classification for lines with multiple factors/single factors.
* Verify 50 char short description for lines with multiple factors / single factors.

### Historical Claims (Out Of Scope)

* Verify PHI is stripped before data sent to DL (Out Of Scope)
* Data validation of historical data (Out of Scope)
* Verify historical claims populated in table DentaLens.dbo.historical\_input. (Not clear on what all fields will be used in the sample historical file, for the purpose of testing)
* Verify frequency of historical claim file transfer.
* Verify historical file processing status is recorded in to MPIEFP.EDI\_PROCESS\_LOG table.
* Verify copy of the same file with date and timestamp is copied to informatica Outbound folder, after historical file processing.
* Verify notifications sent out in case of parsing errors while loading historical claims. (To ask Lakshman to add appropriate email ids for SQA testing?)

### Error Handling

Failures in all of the below services and processes will be checked as part of error handling.

| **SNo** | **UCWClaimTransactionSvc**  **(1)** | **UCWClaimValidationSvc**  **(2)** | **UCWDentaLensAnalyticsSvc**  **(3)** | **DL Procedure**  **g\_00\_checkDataDumpCompletion**  **(4)** | **Expected Behavior** |
| --- | --- | --- | --- | --- | --- |
| 1.1 | Failure– Svc down | - |  |  | Claim put on hold. The release process will resubmit the claim from failure point. |
| 2.1 | Success | Validation Failure  (Will wait for updated requirements) |  |  | Claim put on hold. The release process will resubmit the claim from failure point. |
| 3.1 | Success | Success | Failure - StageClaimForAnalytics |  | Claim put on hold. The release process will resubmit the claim from failure point. |
| 3.2 | Success | Success | Failure – RunAnalyticsForBatch |  | Email alert is generated to support team indicating the failure reason. Claims will be reprocessed with next scheduled control-m job  (email id to be configured in the Ctrl-M scheduler) |
| 3.3 | Success | Success | Failure - RetrieveAnalytics (Unable to update the UCW DB due to some reason) |  | Claim to be put on HOLD. |
| 3.4 | Success | Success | Failure - RetrieveAnalytics (Failure like Connection Time Out) |  | Email alert is generated to support team indicating the failure reason. Claims will be reprocessed with next scheduled control-m job. |
| 3.5 | Success | Success | Invalid result from DentaLens. Anything other than A/D/I |  | Results\_output.is\_processed=-1.  Emails to designated groups.  Claims reprocessed after correction. |
| 4.1 | Success | Success |  | Failure | When procedure re-executed, the claims already copied to output table will not be copied again. |

## Regression Testing

The below existing UCW service is changed as part of the Dentalens project:

* ‘UCWClaimTransactionSvc’.

No regression

## Performance/Volume Testing

Pass / Fail Criteria

## Approval Criteria

All functional and E2E test case execution should be complete in Quality Center.

# Testing Controls & Procedures

## Test Deliverables

* Test Strategy for functional and E2E testing
* Test Cases for functional / E2E testing

## Testing Tasks

* Test Data creation
* Functional Testing
* Regression Testing
* E2E testing

## Key Roles and Responsibilities

| Role | Responsibility |
| --- | --- |
| SQA | * Manmeet Chadha |
| BAT |  |
| EDI |  |
| BA, functional core team | * Gowri Narasimhan |
| Business SMEs |  |
| Project Management | * Mahender Singh |
| Technical Team | * Pramod Paidipelly * Sanjay Ranganathan |

## Resources

Enter text here.

## Schedule

Test planning: 12/3 – 12/11

Test cases: 12/12- 12/21

Execution (Functional and E2E) : 1/2- 1/30

## Testing Success Criteria

### Entrance Criteria

| **Entry Criteria** | **Test Team** | **Technical Team** | **Notes** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

### Exit Criteria

| **Exit Criteria** | **Test Team** | **Technical Team** | **Notes** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Defect Management

All defects will be raised in JIRA, for Version=’DentaLens’

### Defect Management Severity Definitions

1 - Urgent

2 - High

3 - Medium

4 - Low

### Defect Management of Testing Completion

All defects for Version=’DentaLens’ should be Closed or Deferred to a later release, for marking Test Completion.

## Risk Management

### Risks and Assumptions

|  |  |  |
| --- | --- | --- |
| **Risk Number** | **Risk Description** | **Mitigation Steps** |
|  |  |  |

## Progress Reporting

Enter text here.

Environmental Requirements

## Test Environment Provisioning Request Process

* The product ‘xDentalReview’ to be in place in SQA.
* ETL processes to be in place, in SQA.
* The new database instance to be in place on the DL side, for dental claim flow.

# References

| Artifact Name | Location/ Link | Notes |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Glossary

|  |  |
| --- | --- |
| Term | Description |
|  |  |
|  |  |
|  |  |