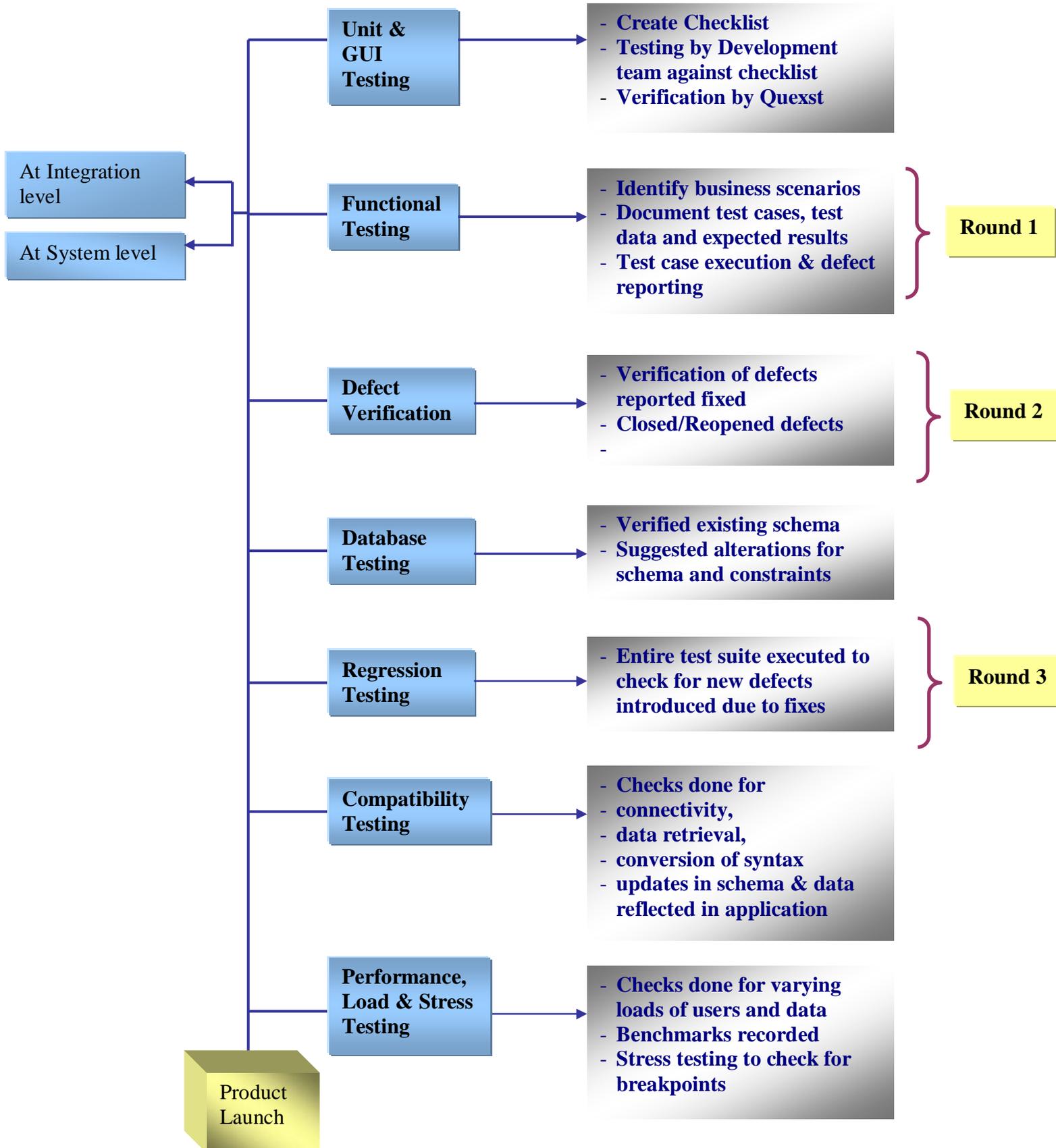


Annexure 1 – Testing Methodologies

The following diagram depicts the testing methodologies used to carry out testing of 1KEY 2.0.1 version.



Detailed Description of the testing methodologies:

The Application is designed using .NET framework, Delphi and report definitions are stored in SQL Express 2005.

1. Unit & GUI Testing

A standard unit level and GUI level checklist was given to the development team. Unit level testing was carried out by the development team against this checklist. Quexst team has verified the same during Functional Testing.

2. Functional Testing

The Functional Testing has covered checking of individual modules, module level integration testing and system testing

Round 1 included –

- **Business Scenarios** identification
- **Test Specification:** documenting test cases and expected result
- **Creation of Test Data**
Since the purpose of the application is to analyze and generate reports from vast amount of data, test data generation was an important activity. Test data was created by using queries and inserting data into various databases/RDBMS.
- **Definition of User rights:**
Defining User rights was a key aspect in this round. Since multiple users can access the application, it becomes necessary to decide the different users, permissions and test the same.
- **Test Execution and Defect reporting:**
Defects were reported on daily basis after tests were executed. The defects were fixed in development area simultaneously.

Round 2 was primarily for defects verification.

All defects reported were verified to ensure their closure.

Round 3 Aiming at defects verification and complete Regression testing

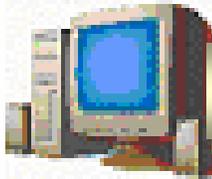
Entire suite was executed to verify if any new defects have been introduced or uncovered as a result of the fixes/changes.

3. Database Testing

Being a Business Intelligence reporting tool, we carried out testing with a specific objective to test existing database design and give appropriate suggestions for improvement. For example: some defects were due to invalid schema, constraints. The same was discussed and reported during database testing.

4. Compatibility Testing

Since the end user can use different RDBMS/DBMS and operating systems, the application was tested for compatibility. This involved setting up the environment with the following combinations:

RDBMS / DBMS Operating Systems				
 Windows XP Family	<p>Across all platforms checked for –</p> <ul style="list-style-type: none"> a. Connectivity b. Data retrieval c. Conversion of RDBMS/DBMS SQL syntax into application Query Designer SQL syntax d. Data updates in the backends being correctly reflected correctly in application e. Update in database schema being reflected correctly in application 			
 Windows 2000 Professional				
 Windows 2003 Server				

It is verified that

- MS Windows 98 users can work with IKEY client application.
- The reports can be generated from MS Excel

5. Performance Testing, Load Testing & Stress Testing

Multiple users are expected to access the application and with varying size of data. Hence, it becomes crucial to test the application performance at peak load and its behavior under stress.

After study of the load testing requirements, it was decided to use a hybrid testing methodology. A lab was setup with:

- 100 clients with 512 MB Ram and Windows XP Professional
- 1 Server with 1 GB Ram and Windows 2003 Server

6. Installation Testing:

For installing and setting up the software it is necessary to follow specific sequential procedure. IKEY installation is verified with the installation 'Readme' document provided along with. This testing covered

- Minimum Hardware requirement specified.
- Required softwares and components are provided in Installation CD.
- Installation procedure is described in sequential order.
- Procedure required for backend connectivity.

7. Single Server User License Application Testing:

This application is designed for a single user.

- One and only one user can be created.
- No user can access the IKEY server from any other client / server.

A number of scenarios were identified, tested (See Annexure 2 for details) and the reports submitted. We created a document which can be served as a benchmark data for this application.

Annexure 2 contains results of all the testing.