# **Product Testing Plan**

V 1.0 11/25/2019



Mohamad Saab Aalem Singh John Gettel Abdul Ahmad

## **Document Version History**

Version	Date	Author	Reviewer	Approved by	Description
0.1	11/10/2019	Abdul Ahmad	N/A	N/A	Created the new document, formatted, wrote starter text in all sections, wrote TC 1, 3
0.2	11/15/2019	Abdul Ahmad	N/A	N/A	Completed section 1 and 2 with all subsections, proofread
0.3	11/21/2019	Abdul Ahmad; John Gettel;	N/A	N/A	Formatting, TOC hyperlinks, wrote section 1.1, 2, reformatted test case boxes
0.4	11/22/2019	Abdul Ahmad; John Gettel; Mohamad Saab;	Abdul Ahmad	N/A	Several test cases with scripts, proofread/reviewed
0.5	11/23/2019	Abdul Ahmad; John Gettel; Aalem Singh;	Abdul Ahmad	N/A	Several test cases with scripts, added section 3, proofread/reviewed
0.6	11/24/2019	Abdul Ahmad; John Gettel;	Abdul Ahmad	N/A	Added remaining functional test cases, completed nonfunctional test cases, wrote section 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, appendix, proofread/reviewed
1.0	11/25/2019	Abdul Ahmad	N/A	John Gettel; Aalem Singh; Mohamad Saab;	TOC links, formatted all sections, page breaks, code snippets at the end, proofread/reviewed

#### **TABLE OF CONTENTS**

1 INTRODUCTION	4
1.1 Purpose of the Product Testing Plan Document	4
1.2 References	
2 FUNCTION TESTING	4
2.1 Test Approach	4
2.2 Steps	4
2.3 Pass/Fail Criteria	4
2.4 Entry/Exit Criteria	4
2.5 Suspension/Resumption Criteria	5
2.6 Test Risks	5
2.7 Expected Results	5
2.8 Priority/Priority Reason	5
2.9 Pre-Condition and Post-Condition	5
2.10 Functional Test Cases	6
3 NON-FUNCTIONAL TESTING	. <b>62</b>
3.1 Non-Functional Testing Approach	. 62
3.2 Non-Functional Testing Pass/Fail Criteria	. 62
3.3 Non-Functional Testing Risks	. 62
3.4 Non-Functional Testing Expected Results	. 62
3.5 Non-Functional Testing Priority/Priority Reason	. 62
3.6 Non-Functional Test Cases	. 63
4 PRODUCT TESTING CERTIFICATION	. 84
APPENDIX	85
APPENDIX A: SQL and Python Testing Scripts	. 85
APPENDIX B: Key Terms	113

#### 1 INTRODUCTION

#### 1.1 PURPOSE OF THE PRODUCT TESTING PLAN DOCUMENT

The purpose of the GM FinTech Product Testing Plan document is to test all the functionality, frontend, database and middle-layer calculations(Python) of the application. The test cases will go through each requirement laid out in the Software Requirements Specification document. The Test Plan document tracks the necessary information required to effectively define the approach to be used in the testing of the project's product. The Test Plan document is created during the Planning Phase of the project. The intended audience of this document includes but not limited to, the project manager, project team, and testing team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input and approval into the testing process is needed.

#### 1.2 REFERENCES

The test cases listed in this document have a direct impact on the data visualized in the Tableau user interface. The use cases listed in the design document are used to reference the test cases that are implemented.

#### 2 FUNCTION TESTING

#### 2.1 TEST APPROACH

Using the testing framework described in this document, the connection to the data source (YAHOO), middle-layer Python processes and database objects will be tested. A walkthrough of the testing process including SQL scripts for creating tables from the database, will be detailed throughout this document.

#### **2.2 STEPS**

Describe the overall approach to be used for the test in a step by step manner.

#### 2.3 PASS/FAIL CRITERIA

If each individual test case matches the desired outcomes, the test is considered successful. If any of the test cases do not meet the predicted criteria, the test case is marked as a failure. A failed test must be reviewed and resolved.

#### 2.4 ENTRY/EXIT CRITERIA

The Entry Criteria of functional testing is completion of Prototype 3. The Exit Criteria of functional testing is the completion of all stated test cases. This is the entry and exit criteria used to start testing and determine when to stop testing.

#### 2.5 SUSPENSION/RESUMPTION CRITERIA

Testing will be suspended when a test case fails or when outside factors like change of product design occur. Testing will resume when these preventative factors are identified and resolved.

#### 2.6 TEST RISKS

All machines utilizing this application must have the same software and connection setup. Any missing component will cause the application to not work as intended. The database must be setup properly, including the order of fields in the table. All hardware and software requirements as listed in the software requirement specifications document must be fulfilled. Availability of a decent internet connection is also required.

#### 2.7 EXPECTED RESULTS

This section describes what is the expected result of the undergoing test. Most results are in the form of a data value return. Some tests are to check Python script functionality and validity.

#### 2.8 PRIORITY/PRIORITY REASON

Priority number and the reason why the test is prioritized at that number.

#### 2.9 PRE-CONDITION AND POST-CONDITION

These are the set of conditions that exist before a test begins. Username and password for the MySQL is created at the time of setup so it could be different for a tester/user than whats listed in this document. This document lists the generic password that a tester can use. Tester's machine's MySQL must be setup with this username and password first otherwise they will have to use their local username and password. This username and password should then be added to PyCharm code file named 'dbEngine.py' in the 'conn-str' variable. Post condition is the application's status and state after the test has completed(success or failure).

#### **2.10 FUNCTIONAL TEST CASES**

ID	TC_1.0		
Test Case Name	Database Engine has the correct MYSQL connection string		
Created By	Abdul Ahmad Date Created 11/10/2019		
Description	Application must use the proper MySQL database connection string.		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in the future, there are no restrictions		
Executed By	Application User, Developer or anyone performing QA Testing		
Preconditions	1 PyCharm is open		
	2 Project is open in PyCharm		
	3 MySQL is working properly		
	4 In PyCharm 'dbEngine.py' file has <b>Username</b> = root		
	5 In PyCharm 'dbEngine.py' file has <b>Password</b> = password		
Postconditions	1 Test is completed successfully		
	2 Middle layer of Python will push data to the MySQL database properly		
	3 Database will be loaded with accurate records		
Test Steps	Following are the steps		
	1 Open PyCharm		
	2 Navigate to Fistertab → F2019 → Unit Tests		
	3 Open TC_1.0.py		
	4 Click on Run menu → Run button		
	5 The test is executed		
<b>Expected Result</b>	Tester should receive one of the following message "incorrect engine" or		
	"correct engine"		
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	<b>Exit:</b> When the test completes or it can be terminated during the run		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database. Extra or less values could cause issues that will		
5 /5 !! 6 !: . !	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: If the message is "correct engine"		
Fail Danner	Failed: If the message is "incorrect engine"		
Fail Reasons	1 Wrong MySQL string in dbEngine.py		
	<ul><li>2 MySQL database is not function properly</li><li>3 MySQL database username or password is incorrect</li></ul>		
Priority			
Filolity	High Priority Reason Important to have the database connection		
	working properly		
Priority Sequence	H-1		
Thority Sequence	11-4		

ID	TC_2.0			
Test Case Name	Correct financial data exchange for fetching base statistics			
Created By	Abdul Ahmad	Date Created	11-10-2019	
Description	Application is designed to use a free financial data exchange. It is setup to			
	use 'YAHOO'. The test needs to va	alidate that 'YAHO	O' is being used for	
	fetching financial records.			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytir	ne in the future, th	nere are no restrictions	
Executed By	Application User, Developer or an			
Preconditions	1 PyCharm is open	, ,		
	2 Project is open in PyCharm	า		
	3 MySQL is working properly	/		
	4 In PyCharm 'dbEngine.py'	file has <b>Username</b>	= root	
	5 In PyCharm 'dbEngine.py'		= password	
Postconditions	<ol> <li>Test will be completed suc</li> </ol>	•		
	2 Middle layer of Python wil	•	•	
	properly using the expecte	_	_	
	3 Database will be loaded w	ith accurate record	ds	
Test Steps				
	1 Open PyCharm			
	2 Navigate to Fistertab → F2019 → Unit Tests			
	3 Open TC_2.0.py  4 Click on Pun monu → Pun button			
	4 Click on Run menu → Run button 5 The test is executed			
Expected Result	Tester should receive one of the following message "incorrect exchange" or			
Expected Result	"correct exchange"			
Test Entry/Exit	<b>Entry</b> : Test can be started at any	time during testing	nhace or later when the	
Criteria	application is deployed. Test itself			
Circuia	<b>Exit:</b> Test can be exited at any tin			
Test Risks	This test itself causes no risk to th			
	existing data in the database. Extr	•	• •	
	need to be corrected by a developer.			
Pass/Fail Criteria	Passed: If the message is "correct exchange"			
	Failed: If the message is "incorrect engine"			
Fail Reasons	1 - Internet connection issues			
	2 – YAHOO exchange is down			
	3 — 'dbEngine.py' or 'DataFetch' python code is corrupted			
Priority	High Priority Reason Availability of free			
	financial data			
			exchange is required	
Priority Sequence	H-2			

ID	TC_3.0		
Test Case Name	No duplicate instrument ids in 'instrumentmaster' table.		
Created By	Abdul Ahmad Date Created 11-21-2019		
Description	No duplicate instrument ids in 'instrumentmaster' table in the database.		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in the future, there are no restrictions		
Executed By	Application User, Developer or anyone performing QA Testing		
Preconditions	<ul> <li>Database is created with MySQL Username = root Password = password</li> <li>Data table exists</li> <li>Data table has values</li> </ul>		
Postconditions	<ul><li>1 No duplicate records exist in the table</li><li>2 Database stays intact</li></ul>		
Test Steps	<ol> <li>Open MySQL WB</li> <li>Browse to Unit Test folder</li> <li>Open TC_3.0.sql</li> <li>Execute the script or press Ctrl+Shift+Enter key on a Windows machine</li> </ol>		
<b>Expected Result</b>	No records should show up in the Result Grid in the MySQL Workbench		
Test Entry/Exit Criteria	Entry: Test can be started at any time during testing phase or later when the application is deployed. Test itself has no direct impact on the working of the application  Exit: Test can be exited at any time during the run or after it finishes		
Test Risks	This test itself causes no risk to the operation of the application or the existing data in the database. Extra or less values could cause issues that will need to be corrected by a developer.		
Pass/Fail Criteria	Passed: No records should show in the Results Grid in MySQL Workbench Failed: If any records show up in the Results Grid, the record will have a value of '2' in the 'cnt' column.		
Fail Reasons	<ul> <li>MySQL database table 'dbo_instrumentmaster' has multiple records for each instrument</li> <li>'dbo_instrumentmaster' table is not setup correctly</li> <li>Manually inserted duplicate values into the 'dbo_instrumentmaster' table</li> </ul>		
Priority	High Priority Reason Duplicate records will throw off all related calculations		
<b>Priority Sequence</b>	H-3		

ID	TC_4.0			
Test Case Name	No duplicate values/records in future forecast for each symbol			
Created By	Abdul Ahmad	<b>Date Created</b>	11-21-2019	
Description	No duplicate values/records in future forecast for each symbol in the			
	database			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restrictions			
Executed By	Application User			
Preconditions	1 MySQL <b>Username</b> = root <b>Password</b> :	= password		
	2 Database is created			
	3 Data table 'dbo_algorithmforecast' ex	xists		
	4 Data table is not empty			
Postconditions	1 – No duplicate records exist in the tal	ole		
	2 – Database stays intact			
Test Steps	1 – Open MySQL WorkBench			
	2 – Browse to Unit Test folder			
	3 – Open TC_4.0.sql			
	4 – Execute the script or press Ctrl+Sh	ift+Enter key on a V	Vindows machine	
Expected Result	5 records with each showing 'cnt' valu	e of 10 on each rec	ord in MySOI	
Expected Result	5 records with each showing 'cnt' value of 10 on each record in MySQL WorkBench's Result Grid			
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the			
Criteria	application is deployed. Test itself has no direct impact on the working of the			
	application			
	Exit: Test can be exited at any time during the run or after it finishes			
Test Risks	This test itself causes no risk to the operation of the application or the			
	existing data in the database. Extra val			
	to be corrected by a developer.			
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench shows 5 re	ecords and 'cnt'	
	value of 10 for each record			
	Failed: The Result Grid in MySQL Wor	rkBench shows mor	e than 5 records or	
	'cnt' value not equal to 10 for each record			
Fail Reasons	MySQL database table 'dbo_algorithmforecast' has multiple records			
	for each instrument for future dates			
	2 'dbo_algorithmforecast' table is not setup correctly			
Priority	Medium	<b>Priority Reason</b>	Extra forecast	
			values will not	
			impact other	
	calculations			
<b>Priority Sequence</b>	M-1			

ID	TC_5.0			
Test Case Name	No duplicate values/records in past forecast for each symbol			
Created By	Abdul Ahmad Date Created 11-21-2019			
Description	No duplicate values/records in past forecast for each symbol in the databse			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ired		
<b>Executed By</b>	Application User			
Preconditions	1 Database is created with MySQL <b>Username</b> = root <b>Password</b> = password			
	2 Data table exists			
	3 Data table has values			
Postconditions	1 No duplicate records exist in the tabl	e		
	2 Database stays intact			
Test Steps	1 Open MySQL WorkBench			
	2 Browse to Unit Test folder			
	3 Open TC_5.0.sql			
	4 Execute the script or press Ctrl+Shift	+Enter key on a Wi	ndows machine	
Expected Result	5 records with each showing 'cnt' value	e of 751 on each re	ecord in MySQL	
	WorkBench's Result Grid			
Test Entry/Exit	<b>Entry</b> : Test can be started at any time during testing phase or later when the			
Criteria	application is deployed. Test itself has no direct impact on the working of the			
	application			
Test Risks	Exit: Test can be exited at any time during the run or after it finishes  This test itself causes no risk to the operation of the application or the			
Test Kisks	existing data in the database. Extra or	• •		
	need to be corrected by a developer.	iess values coulu ca	duse issues that will	
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkRench shows 5 re	ecords and 'cnt'	
r assyr an enteria	value of 751 for each record	INDEFICIT SHOWS 5 TO	cords and the	
	Failed: The Result Grid in MySQL Wor	kBench shows mo	re than 5 records or	
	'cnt' value not equal to 751 for each re			
Fail Reasons	1 MySQL database table 'dbo alg		as multiple records	
	for each instrument for future		'	
	2 'dbo algorithmforecast' table is not setup correctly			
Priority	Medium	<b>Priority Reason</b>	Extra forecast	
			values will not	
	impact other			
	calculations			
Priority Sequence	M-2			

ID	TC_6.0		
Test Case Name	Future Forecast should exist for each symbol		
Created By	Mohamad Saab	Date Created	11-21-2019
Description	There should be forecasted close price data for future dates (1 or more days		
	past the day the test is run) for each sy	mbol in the tool	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application is opened		
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Us</b>		assword = password
	2 Run project using DataMain in Pyth		
	3 Data table has values for symbols a	·	
Postconditions	No duplications of forecast data past o	r future in databas	Se Se
Test Steps	Open MySQL workbench		
	2. Browse to File > Open SQL Scrip		
	3. Locate "TC_6.0" in SQL folder in	• •	pen
	4. Run Script by pressing Execute	, , , ,	
Funcated Beaut	5. Note results in execution window		
Expected Result	Table showing: 1+ instrument ID; forecasted prices(avg); next day as a future date (day after testers present day); last day in the future; and 1+ count for		
	future dates		
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the		
Criteria			
Circuia	application is deployed. Test itself has no direct impact on the working of the application		
	Exit: Test can be exited at any time du	ring the run or afte	er it finishes
Test Risks	No risk to project, SQL script is indeper	ndent of project	
Pass/Fail Criteria	Passed: Table showing future dates an		
	Failed: Table returns empty or with no future dates		
Fail Reasons	1 Not connected to database		
	Database doesn't produce future data		
Priority	High	<b>Priority Reason</b>	<b>Duplication of</b>
			master symbols
	could duplicate al		
	calculations		
<b>Priority Sequence</b>	H-2		

ID	TC_7.0			
Test Case Name	Future Forecast should exist for each algorithm			
Created By	Mohamad Saab	<b>Date Created</b>	11-21-2019	
Description	There should be forecasted close price	data for future dat	es (1 or more days	
	past the day the test is run) for each alg	orithm in the tool		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application is opened			
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>User</b>		word = password	
	2 Run project using DataMain in Python			
	3 Database is created			
	4 Data table has values for symbols and	•		
Postconditions	No duplications of forecast data past or	future in database	e	
Test Steps	6. Open mySQL workbench			
	7. Browse to File>Open SQL Script:			
	8. Locate "TC_7.0" in SQL folder in project folders, Open			
	9. Run Script by pressing Execute (lightning symbol)  10. Note results in execution window			
Expected Result	Table showing: 1+ algorithm name; forecasted prices(avg); next day as a			
Expected Result	future date (day after testers present day); last day in the future; and 1+			
	count for future dates			
Test Entry/Exit	<b>Entry:</b> Test can be started at any time during testing phase or later when the			
Criteria	application is deployed. Test itself has r			
	application	•	Ü	
	<b>Exit</b> : Test can be exited at any time dur		r it finishes	
Test Risks	No risk to project, SQL script is independent of project			
Pass/Fail Criteria	Passed: Table showing future dates an			
	Failed: Table returns empty or with no future dates			
Fail Reasons	1 Not connected to database			
Driority	2 Database doesn't produce future data			
Priority	High Priority Reason Duplicate symbols			
	will duplicate all calculations			
Priority Sequence	H-2		calculations	
Thority sequence	11 6			

ID	TC_8.0		
<b>Test Case Name</b>	Past forecast values must exist for last 3 years until today for each symbol		
Created By	Abdul Ahmad	Date Created	11-21-2019
Description	Past forecast values must exist for last	3 years until today	for each symbol in
	the database		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in	the future, there a	re no restrictions
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Username</b> = root <b>Password</b> = password		
	2 Data table exists		
	3 Data table has values		
Postconditions	1 Records exist in the table prior to too	day's date	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_8.0.sql		1
E consideration	4 Execute the script or press Ctrl+Shift	•	
Expected Result	5 records with each showing 'date' value of 751 in MySQL WorkBench's		
Took Finking / Finish	Result Grid		
Test Entry/Exit Criteria	<b>Entry:</b> Test can be started at any time during testing phase or later when the		
Criteria	application is deployed. Test itself has no direct impact on the working of the		
	application  Exit: Test can be exited at any time during the run or after it finishes		
Test Risks	This test itself causes no risk to the ope		
rest rusks	existing data in the database. Extra or	• •	
	need to be corrected by a developer.	icos varaes coura co	ause issues that will
Pass/Fail Criteria	Passed : The Result Grid in MySQL Wo	rkBench shows valu	ue of 5 in the 'instid'
	column and 'date' value of 751		
	Failed: The Result Grid in MySQL Wor	kBench shows moi	re than 1 record or
	'date' value not equal to 751		
Fail Reasons	3 MySQL database table 'dbo_algorithmforecast' has missing records		
	for the past		
	4 'dbo_algorithmforecast' table is not setup correctly		
Priority	High Priority Reason Past forecasted		
Thornty	'''8''	Triority Reason	values are very
			important
Priority Sequence	H-6		portant
or.ty ocquerice			

ID	TC_9.0		
Test Case Name	Past forecast values must exist for last 3 years until today for each algorithm		
Created By	Abdul Ahmad	<b>Date Created</b>	11-21-2019
Description	Past forecast values must exist for last	3 years until today	for each algorithm
	in the database		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in	the future, there a	re no restrictions
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Username</b> = root <b>Password</b> = password		
	2 Data table exists		
	3 Data table has values		
Postconditions	1 Records exist in the table prior to too	day's date	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_9.0.sql		
	4 Execute the script or press Ctrl+Shift	•	
Expected Result	'algo' value must be 6 and 'date' value should be 751 in MySQL WorkBench's		
/	Result Grid		
Test Entry/Exit	<b>Entry</b> : Test can be started at any time during testing phase or later when the		
Criteria	application is deployed. Test itself has no direct impact on the working of the		
	application  Exit: Test can be exited at any time during the run or after it finishes		
Test Risks	This test itself causes no risk to the ope		
1 CSC INISKS	existing data in the database. Extra or	• •	
	need to be corrected by a developer.	icss values could co	dase issues that will
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench shows vali	ue of 6 in the 'algo'
1 455,1 411 61114	column and 'date' value of 751	. KBerrerr Smotts var.	ac 0. 0 the a.go
	Failed: The Result Grid in MySQL Wor	kBench shows moi	re than 1 record or
	'date' value not equal to 751		
Fail Reasons	1 MySQL database table 'dbo_algorith	mforecast' has mis	sing records for the
	past		
	2 'dbo_algorithmforecast' table is not setup correctly		
<b>D.</b>			
Priority	High	Priority Reason	Displaying past
			forecast values is
Driority Convenes	U 7		required
<b>Priority Sequence</b>	H-7		

ID	TC_10.0		
Test Case Name	Ten future days from today with no prediction error for each symbol		
Created By	Abdul Ahmad	Date Created	11-21-2019
Description	Ten future days from today with no pr	ediction error for e	ach symbol in the
	database. Future dates have no 'close'	value so we can't	calculate our Mean
	Base Error rate.		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in	the future, there a	re no restrictions
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password
	2 Data table exists		
	3 Data table has values		
Postconditions	1 Records exist in the table after today	r's date	
_	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_10.0.sql	. F	
Francisco di Documb	4 Execute the script or press Ctrl+Shift+Enter key on a Windows machine		
Expected Result	'date' value must be 10 and 'prederror' should be 0 in MySQL WorkBench's Result Grid		
Test Entry/Exit			
Criteria	<b>Entry:</b> Test can be started at any time during testing phase or later when the		
Criteria	application is deployed. Test itself has no direct impact on the working of the application		
	Exit: Test can be exited at any time during the run or after it finishes		
Test Risks	This test itself causes no risk to the ope		
	existing data in the database. Extra or	•	
	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench shows valu	ue of 10 in the
	'date' column and 0 in the 'prederror'		
	Failed: The Result Grid in MySQL Wor	kBench shows mor	re than 1 record or
	'date' value not equal to 10		
Fail Reasons	1 MySQL database table 'dbo_algorithmforecast' has missing records for the		
	future		
	2 'dbo_algorithmforecast' table is not setup correctly		
Priority	Medium	Priority Reason	Future forecast
	should not have		
			Mean Base Error
<b>Priority Sequence</b>	M-3		

ID	TC_11.0		
Test Case Name	Ten future days from today with no prediction error for each algorithm		
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019
Description	Ten future days from today with no pr	ediction error for e	ach algorithm in the
	database		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in	the future, there a	re no restrictions
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password
	2 Data table exists		
	3 Data table has values		
Postconditions	1 Records exist in the table after today	r's date	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_11.0.sql		
	4 Execute the script or press Ctrl+Shift	•	
Expected Result	'date' value must be 10 and 'prederror	r' should be 0 in My	/SQL WorkBench's
/	Result Grid	1 1 1 1	
Test Entry/Exit	<b>Entry:</b> Test can be started at any time		
Criteria	application is deployed. Test itself has	no direct impact or	n the working of the
	application		
Test Risks	<b>Exit :</b> Test can be exited at any time during the run or after it finishes  This test itself causes no risk to the operation of the application or the		
i est nisks	existing data in the database. Extra or	• •	
	need to be corrected by a developer.	icss values could co	dusc issues that will
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench shows valu	ue of 10 in the
1 dosy i dii circeria	'date' column, 5 in the 'algo' column a		
	Failed: The Result Grid in MySQL Wor	•	
	'date' value not equal to 10		
Fail Reasons	1 MySQL database table 'dbo_algorith	mforecast' has mis	sing records for the
	future		0
	2 'dbo_algorithmforecast' table is not	setup correctly	
Priority	Medium	<b>Priority Reason</b>	Algorithm future
			forecast values
			should not have
			Mean Base Error
<b>Priority Sequence</b>	M-4		

ID	TC_12.0		
Test Case Name	Nine tables exist in the database		
Created By	Abdul Ahmad	Date Created	11-22-2019
Description	Nine tables exist in the database. These are the main required tables.		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in	the future, there a	re no restrictions
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b> 2 Data table exists	ername = root Pas	sword = password
Postconditions	Tables stay intact in the databa     Database stays intact	ise	
Test Steps	1 Open MySQL WorkBench		
rest steps	2 Browse to Unit Test folder		
	3 Open TC 12.0.sql		
	4 Execute the script or press Ctrl+Shift	+Enter key on a Wi	ndows machine
Expected Result	'cnt' value must be 9 in MySQL WorkB	ench's Result Grid	
Test Entry/Exit	Entry: Test can be started at any time		se or later when the
Criteria	application is deployed. Test itself has	no direct impact or	n the working of the
	application.		
	Exit: Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the op-	• •	
	existing data in the database. Extra or	less values could ca	ause issues that will
- /- !! o !: !	need to be corrected by a developer.	15 1 1	501 11 / 11
Pass/Fail Criteria	<b>Passed :</b> The Result Grid in MySQL Wo column.	rkBench shows valu	ue of 9 in the 'cnt'
	Failed: The Result Grid in MySQL Wor	rkRanch shaws loss	or more than 0 in
	the 'cnt' column	IKDEHCH SHOWS IESS	of filore than 9 in
Fail Reasons	1 MySQL database has missing or extra	a tables	
	2 Database is not setup correctly		
	2 Database is not setup correctly		
Priority	High	<b>Priority Reason</b>	Tables are
			required for the
			application
			middle-layer and
			front-end to work
Priority Sequence	H-4		

ID	TC_13.0		
Test Case Name	ARIMA has first 9 days blank		
Created By	Abdul Ahmad	Date Created	11-22-2019
Description	ARIMA has first 9 days blank in the tab	le	
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in the future, there are no restrictions		
Executed By	Application User		
Preconditions	Database is created with MySQL <b>Use</b> Data tables exists	rname = root Pas	sword = password
Postconditions	<ul><li>1 Tables stay intact in the databa</li><li>2 Database stays intact</li></ul>	se	
Test Steps	1 Open MySQL WorkBench 2 Browse to Unit Test folder 3 Open TC_13.0.sql 4 Execute the script or press Ctrl+Shift+Enter key on a Windows machine		
Expected Result	'forecastcloseprice' value must be blar WorkBench's Result Grid	nk for first 10 days ı	records in MySQL
Test Entry/Exit Criteria	<b>Entry:</b> Test can be started at any time application is deployed. Test itself has	• • •	
	application. <b>Exit:</b> Test can be exited at any time du	iring the run or afte	er it finishes.
Test Risks	This test itself causes no risk to the ope		
	existing data in the database. Extra or	less values could ca	ause issues that will
	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench shows bla	nk values in the first
	10 days records in the 'forecastclosepr		
	Failed: The Result Grid in MySQL Wor		
Fail Reasons	1 Python script did not pipe prop		nto the table
	2 Database is not setup correctly		
	3 'instrumentstatistics' table is p		
Priority	High	Priority Reason	First 9 days are
			used to feed for
			the forecast for
Driority Convence	H-5		next day
Priority Sequence	п-э		

ID	TC_14.0			
Test Case Name	Random Forest has first 9 days blank			
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019	
Description	Random Forest has first 9 days blank			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restrictions			
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b> 2 Data tables exists	rname = root Pas	sword = password	
	2 Data tables exists			
Postconditions	1 Tables stay intact in the databa	se		
	2 Database stays intact			
Test Steps	1 Open MySQL WorkBench			
rest steps	2 Browse to Unit Test folder			
	3 Open TC_14.0.sql			
	4 Execute the script or press Ctrl+Shift	+Enter key on a Wi	ndows machine	
	' '	,		
<b>Expected Result</b>	'forecastcloseprice' value must be blar	nk for first 10 days i	records in MySQL	
	WorkBench's Result Grid			
Test Entry/Exit	1	Entry: Test can be started at any time during testing phase or later when the		
Criteria	application is deployed. Test itself has	no direct impact or	n the working of the	
	application.			
	Exit: Test can be exited at any time du			
Test Risks	This test itself causes no risk to the op			
	existing data in the database. Extra or	less values could ca	ause issues that will	
Dece/Feil Cuiteuie	need to be corrected by a developer.	عاما ويردواه والمراد	alevalvas in the first	
Pass/Fail Criteria	<b>Passed</b> : The Result Grid in MySQL Wo 10 days records in the 'forecastclosepr		nk values in the first	
	Failed: The Result Grid in MySQL Wor		alank values	
Fail Reasons	3 Python script did not pipe prop			
Tun Reasons	4 Database is not setup correctly		nto the table	
	5 'instrumentstatistics' table is p		ears of values	
Priority	High	Priority Reason	First 9 days are	
		•	used to feed for	
			the forecast for	
			next day	
<b>Priority Sequence</b>	H-6			

ID	TC_15.0		
Test Case Name	No missing values in the 'close' or 'date' column in the 'instrumentstatistics		
	'table		
Created By	Abdul Ahmad	Date Created	11-22-2019
Description	No missing values in the 'close' or 'date	e' column in the 'in	strumentstatistics'
	table		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	ired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pass	sword = password
	2 Data table exists		
Postconditions	1 Tables stay intact in the database		
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	1 Open TC_15.0.sql		
	2 Execute the script or press Ctrl+Shi		
Expected Result	No results should appear in MySQL Wo		
Test Entry/Exit	<b>Entry</b> : Test can be started at any time		
Criteria	application is deployed. Test itself has	no direct impact or	i the working of the
	application.		
Test Risks	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.  This test itself causes no risk to the operation of the application or the		
rest risks	existing data in the database. Extra or less values could cause issues that will		
	need to be corrected by a developer.	iess values could ca	iuse issues that will
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkRench shows no	results returns
r assyr an erreeria	nothing.	TREETICH SHOWS HO	results, returns
	Failed: The Result Grid in MySQL Wor	kBench shows a re	cord
Fail Reasons	1 YAHOO exchange had issues		
	2 DataMain did not run properly		
	3 Python script did not pipe value	es properly into the	table
	4 Database is not setup correctly		
Priority	High	<b>Priority Reason</b>	Each tuple should
			have atomic and
			accurate values
<b>Priority Sequence</b>	H-7		

ID	TC_16.0					
Test Case Name	BuySell signals must exist for all records for past 3 years, values of -1,0,1 for					
	each day					
Created By	Abdul Ahmad	Abdul Ahmad Date Created 11-22-2019				
Description	BuySell signals must exist for all records for past 3 years, values of -1,0,1 for					
	each day.					
Tester	Data Analyst or Developer					
Test Frequency	Anytime the application testing is requ	iired				
Executed By	Application User					
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password			
	2 Data table exists					
Postconditions	1 Tables stay intact in the databa	ise				
	2 Database stays intact					
Test Steps	1 Open MySQL WorkBench					
	2 Browse to Unit Test folder					
	3 Open TC_16.0.sql	-1.c				
	4 Execute the script or press Ctrl	+Shift+Enter key or	a Windows			
E control Book In	machine	Danaha ida Giranda				
Expected Result	Results should appear in MySQL WorkBench with Signal count					
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the					
Criteria	application is deployed. Test itself has no direct impact on the system.					
Test Risks	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.  This test itself causes no risk to the operation of the application or the					
Test Nisks	<u> </u>					
	existing data in the database. Extra or less values could cause issues that will need to be corrected by a developer.					
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkRench shows last	· 3 years dates with			
1 ussy i un circeria	each date and count of signals betwee		. 5 years dates with			
	Failed: The Result Grid in MySQL Wor		record where			
	'signals' column shows values <1 or >3	<del>-</del> -				
Fail Reasons	1 DataMain did not run prope					
	2 Python script did not pipe v	•	the table			
	3 BuySell strategy did not wo					
	4 Database is not setup corre	• • •				
Priority	High	<b>Priority Reason</b>	BuySell signals are			
			important to			
			many other			
	calculations					
<b>Priority Sequence</b>	H-8					

ID	TC_17.0		
Test Case Name	BuySell signals must exist for each stra	tegy	
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019
Description	BuySell signals must exist for each stra	tegy	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password
	2 Data table exists		
Postconditions	1 Tables stay intact in the database		
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_17.0.sql		
	4 Execute the script or press Ctrl+Sh	The state of the s	
Expected Result	Results should appear in MySQL Work		
Test Entry/Exit	Entry: Test can be started at any time		
Criteria	application is deployed. Test itself has	no direct impact or	n the working of the
	application.		
	Exit: Test can be exited at any time du		
Test Risks	This test itself causes no risk to the op	• •	
	existing data in the database. Extra or	less values could ca	ause issues that will
Dans/Fail Cuitania	need to be corrected by a developer.	ul.Danala alaanna laat	. 2
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo		•
	each date and count of signals betwee  Failed: The Result Grid in MySQL Wor		
	'signals' column shows values <1 or >3	•	record where
Fail Reasons	1 DataMain did not run prope		
Tall Neasons	2 Python script did not pipe v	•	the table
	3 BuySell strategy did not wo		the table
	4 Database is not setup correctly		
Priority	High	Priority Reason	Each strategy
,			implemented
			must have Buy(1),
			Sell(-1) or Hold(0)
Priority Sequence	H-9		

ID	TC_18.0		
Test Case Name	BuySell signals must exist for each symbol		
Created By	Abdul Ahmad	Date Created	11-22-2019
Description	BuySell signals must exist for each sym	bol	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	ired	
Executed By	Application User		
Preconditions	<ol> <li>Database is created with MySQ</li> </ol>	L <b>Username</b> = root	Password =
	password		
	2 Data table exists		
Postconditions	1 Tables stay intact in the databa	se	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_18.0.sql		147
	4 Execute the script or press Ctrl-	+Shift+Enter key on	a Windows
Forested Descrip	machine	مريامين والخارين وام مرم ي	_
Expected Result	No result should appear in MySQL Wo		
Test Entry/Exit Criteria	<b>Entry:</b> Test can be started at any time		
Criteria	application is deployed. Test itself has	no direct impact or	i the working of the
	application. <b>Exit:</b> Test can be exited at any time du	ring the run or afte	ar it finishes
Test Risks	This test itself causes no risk to the ope		
rest risks	existing data in the database. Extra or		
	need to be corrected by a developer.	iess values could co	idse issues that will
Pass/Fail Criteria	Passed : The Result Grid in MySQL Wo	rkBench return no	record
,	Failed: The Result Grid in MySQL Wor		
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value	es properly into the	table
	3 BuySell strategy did not work p	roperly	
	4 Database is not setup correctly		
Priority	High	<b>Priority Reason</b>	Each symbol must
			have Buy(1), Sell(-
			1) or Hold(0)
			signal
Priority Sequence	H-10		

ID	TC_19.0		
Test Case Name	Six algorithm codes exist in 'algorithmmaster' table		
Created By	Abdul Ahmad	Date Created	11-22-2019
Description	Six algorithm codes exist in 'algorithm	master' table	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	<ol> <li>Database is created with MySO</li> </ol>	L <b>Username</b> = root	Password =
	password		
	2 Data table exists		
Postconditions	1 Tables stay intact in the databa	ise	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_19.0.sql	al 16. =	
	4 Execute the script or press Ctrl	+Shift+Enter key on	i a Windows
	machine	1 11 1 11	1 1
Expected Result	6 rows are returned in MySQL WorkBe		
Test Entry/Exit	<b>Entry</b> : Test can be started at any time		
Criteria	application is deployed. Test itself has no direct impact on the working of the		
	application. <b>Exit:</b> Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the operation of the application or the		
Test Misks	existing data in the database. Extra or		
	need to be corrected by a developer.	icss values could co	ause issues that will
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns 6 r	records
	Failed: The Result Grid in MySQL Wor		
	,,,,		
Fail Reasons	1 DataMain did not run prope	erly	
	2 Python script did not pipe v	•	the table
	3 BuySell strategy did not wo	rk properly	
	4 Database is not setup corre	ectly	
Priority	Medium	<b>Priority Reason</b>	Algorithmcodes
			are picked up and
			piped to this table
Priority Sequence	M-5		

ID	TC_20.0		
Test Case Name	'Instrumentmaster' must have only 5 stock symbols		
Created By	Abdul Ahmad	Date Created	11-22-2019
Description	'Instrumentmaster' must have only 5 s	stock symbols	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	<ul><li>1 Database is created with MySQL U</li><li>2 Data table exists</li></ul>	sername = root Pa	assword = password
Postconditions	<ul><li>1 Tables stay intact in the databa</li><li>2 Database stays intact</li></ul>	ise	
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_20.0.sql		
	4 Execute the script or press Ctrl- machine	+Shift+Enter key or	n a Windows
Expected Result	5 rows are returned in MySQL WorkBe	nch with instrume	nt values
Test Entry/Exit	Entry: Test can be started at any time	during testing phas	se or later when the
Criteria	application is deployed. Test itself has	no direct impact or	n the working of the
	application.		
	Exit: Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the op	• •	
	existing data in the database. Extra or	less values could ca	ause issues that will
Pass/Fail Criteria	need to be corrected by a developer.  Passed: The Result Grid in MySQL Wo	rkBonch roturns E i	rocards
Pass/Faii Criteria	Failed: The Result Grid in MySQL Wo		
	Tanea . The Result Grid III Mysiqe Wor	RDeficit shows <> c	71000143
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value	es properly into the	e table
	3 BuySell strategy did not work p	roperly	
	4 Database is not setup correctly	,	
Priority	High	<b>Priority Reason</b>	Unique and
			required stock
			symbols are used
			in all calculations
			and
Drionity Converse	Ц 11		measurements
Priority Sequence	H-11		

ID	TC_21.0				
Test Case Name	'Instrumentstats' column named 'high' has values >= 'low', 'open', 'close'				
	columns on all records				
Created By	Abdul Ahmad Date Created 11-23-2019				
Description	'Instrumentstats' column named 'high' has values >= 'low' , 'open', 'close'				
	columns on all records				
Tester	Data Analyst or Developer				
Test Frequency	Anytime the application testing is requ	ired			
Executed By	Application User				
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	assword = password		
	2 Data table exists				
	3 All the fields in the table contain va				
Postconditions	1 Tables stay intact in the databa	se			
<b>T</b> 10:	2 Database stays intact				
Test Steps	1 Open MySQL WorkBench				
	2 Browse to Unit Test folder				
	3 Open TC_21.0.sql 4 Execute the script or press Ctrl-	. Shift + Entar kay an	a Windows		
	machine	rannit Enter key on	i a wiiiuows		
Expected Result	No results are returned in MySQL Wor	kRench with instru	ment values		
Test Entry/Exit	<b>Entry:</b> Test can be started at any time				
Criteria	application is deployed. Test itself has				
	application.	a cot pact o.	The Working of the		
	<b>Exit:</b> Test can be exited at any time du	ring the run or afte	er it finishes.		
Test Risks	This test itself causes no risk to the operation of the application or the				
	existing data in the database. Extra or	less values could ca	ause issues that will		
	need to be corrected by a developer.				
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	records		
	Failed: The Result Grid in MySQL Wor	kBench shows 1 or	more records		
Fail Reasons	1 DataMain did not run prope	•			
	2 Python script did not pipe v		the table		
	3 YAHOO exchange encounte				
B.C. CI	4 Database is not setup corre	•	Wallahaa		
Priority	High	Priority Reason	Valid base		
			statistics are vital to all the		
			calculations		
Priority Sequence	H-11		calculations		
Filolity Sequence	U-TT				

ID	TC_22.0		
Test Case Name	Instrumentstats column names 'low' h	as values <= high, o	ppen, close columns
	on all records		
Created By	Abdul Ahmad	Date Created	11-23-2019
Description	Instrumentstats column names 'low' h	as values <= high, o	ppen, close columns
	on all records		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Us</b>	ername = root Pa	<b>ssword</b> = password
	2 Data table exists		
	3 All the fields in the table contain va		
Postconditions	1 Tables stay intact in the databa	ise	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_22.0.sql	7. F. J.	Park a sanakita
Formanda di Dancolti	4 Execute the script or press Ctrl+Shif	· · · · · · · · · · · · · · · · · · ·	
Expected Result	No results are returned in MySQL Wor		
Test Entry/Exit Criteria	<b>Entry:</b> Test can be started at any time		
Criteria	application is deployed. Test itself has no direct impact on the working of the		
	application.  Exit: Test can be exited at any time du	iring the run or afte	ar it finiches
Test Risks	This test itself causes no risk to the op-		
Test Misks	existing data in the database. Extra or		
	need to be corrected by a developer.	icss values could et	dase issues that will
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	records
	Failed: The Result Grid in MySQL Wor		
	Sometimes due to decimal point a valu		
	to be incorrect but the analyst can visu	ually inspect it to ch	neck for validity.
Fail Reasons	1 DataMain did not run prope	erly	
	2 Python script did not pipe v	alues properly into	the table
	3 YAHOO exchange encounte	ered an issue	
	4 Database is not setup corre	ctly	
Priority	High	<b>Priority Reason</b>	Valid base
			statistics are vital
			to all the
			calculations
Priority Sequence	H-13		

ID	TC_23.0		
Test Case Name	Instrumentstats column 'volume' values should always be >=0		
Created By	Abdul Ahmad Date Created 11-23-2019		
Description	Instrumentstats column 'volume' values should always be >=0, volume is		
	never negative in stock trading.		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	assword = password
	2 Data table exists		
	3 All the fields in the table contain va		
Postconditions	1 Tables stay intact in the databa	se	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_23.0.sql		
	4 Execute the script or press Ctrl+Shift+Enter key on a Windows		
E consideration	machine	l Danielo - Pilo Parie	
Expected Result	No results are returned in MySQL Wor		
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the		
Criteria	application is deployed. Test itself has no direct impact on the working of the		
	application. <b>Exit:</b> Test can be exited at any time during the run or after it finishes.		
Test Risks	-		
Test Misks	This test itself causes no risk to the operation of the application or the existing data in the database. Extra or less values could cause issues that will		
	need to be corrected by a developer.	iess values coula co	dase issues that will
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	records
1 455,1 411 61116114	Failed: The Result Grid in MySQL Wor		
	,,,,		
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value	es properly into the	e table
	3 YAHOO exchange encountered an issue		
	4 Database is not setup correctly		
Priority	High	<b>Priority Reason</b>	Valid base
			statistics are vital
			to all the
			calculations
Priority Sequence	H-14		

ID	TC_24.0			
Test Case Name	Instrumentstats must have 3 years of data till today, going back 3 years from			
	today.			
Created By	Abdul Ahmad Date Created 11-23-2019			
Description	Instrumentstats must have 3 years of	of data till today, goin	g back 3 years from	
	today			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is re	quired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL	Username = root Pa	ssword = password	
	2 Data table exists			
	3 All the fields in the table contain			
Postconditions	1 Tables stay intact in the data	base		
Took Change	2 Database stays intact			
Test Steps	1 Open MySQL WorkBench 2 Browse to Unit Test folder			
	- '	trl±Shift±Entar kay or	n a DC	
Expected Result	4 Execute the script or press Ctrl+Shift+Enter key on a PC 'startdate' must return a date that is 3 years ago from today and 'enddate'			
Expected Result	must return a date that is today/last business day if you are running it on the			
	weekend in MySQL WorkBench with instrument values			
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the			
Criteria	application is deployed. Test itself h			
	Exit: Test can be exited at any time during the run or after it finishes.			
Test Risks	This test itself causes no risk to the operation of the application or the			
	existing data in the database. Extra or less values could cause issues that will			
	need to be corrected by a developer.			
Pass/Fail Criteria	Passed: The Result Grid in MySQL WorkBench returns minimum and			
	maximum dates that comply with 3 year range			
	Failed: The Result Grid in MySQL V	orkBench shows valu	ues that are either	
	over the 3 year limit or under			
Fail Reasons	1 DataMain did not run prope			
	2 Python script did not pipe va		e table	
	3 YAHOO exchange encountered an issue			
Priority	4 Database is not setup correctly  High Priority Reason 3 years data is a			
Filolity	High	<b>Priority Reason</b>	requirement	
Priority Sequence	H-15		requirement	
. Hority sequence	11 13			

ID	TC_25.0			
Test Case Name	Datedim must contain 50 future days from today			
Created By	Abdul Ahmad Date Created 11-23-2019			
Description	Datedim must contain 50 future days f	rom today		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ired		
Executed By	Application User			
Preconditions	1 Database is created with MySQ	L <b>Username</b> = root	Password =	
	password			
	2 Data table exists			
	3 All the fields in the table contai			
Postconditions	1 Tables stay intact in the databa	se		
<b>=</b>	2 Database stays intact			
Test Steps	1 Open MySQL WorkBench			
	2 Browse to Unit Test folder 3 Open TC 25.0.sql			
	. – .	+Shift+Entor kov on	a PC	
Expected Result	4 Execute the script or press Ctrl+Shift+Enter key on a PC  'daysdiff' must return a value of 50 in MySQL WorkBench			
Test Entry/Exit	·	•		
Criteria	<b>Entry</b> : Test can be started at any time during testing phase or later when the application is deployed. Test itself has no direct impact on the system.			
	<b>Exit :</b> Test can be exited at any time during the run or after it finishes.			
Test Risks	This test itself causes no risk to the operation of the application or the			
	existing data in the database. Extra or less values could cause issues that will			
	need to be corrected by a developer.			
Pass/Fail Criteria	Passed: The Result Grid in MySQL WorkBench returns a single row with			
	'daysdiff' column containing value of 5	0		
	Failed: The Result Grid in MySQL Wor	kBench return a va	lue other than 50 in	
	'daysdiff' column			
Fail Reasons	1 DataMain did not run properly			
	2 Python script did not pipe value			
	3 DateFetch script function get_calendar() had issues			
	4 DateDim table is not setup properly			
Priority	5 Database is not setup correctly <b>High</b>	Priority Reason	Future dates are	
Filolity	nigii	Priority Reason	important to	
			show forecasts	
			and predictions	
Priority Sequence	H-16		•	

ID	TC_26.0		
Test Case Name	Datedim must contain last 3 years of dates, up till today		
Created By	Abdul Ahmad	Date Created	11-23-2019
Description	Datedim must contain last 3 years of d	ates, up till today	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	ired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	<b>assword</b> = password
	2 Data table exists		
	3 All the fields in the table contain va		
Postconditions	1 Tables stay intact in the databa	se	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_26.0.sql	. Ch:ft . Fatou loss on	a DC
Eveneted Decult	4 Execute the script or press Ctrl-	•	
Expected Result	'daysdiff' must return a value of 1096	•	
Test Entry/Exit Criteria	<b>Entry:</b> Test can be started at any time		
Criteria	application is deployed. Test itself has no direct impact on the working of the application.		
	Exit: Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database. Extra or less values could cause issues that will		
	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns a s	single row with
	'daysdiff' column containing value of 1	.096	_
	Failed: The Result Grid in MySQL Wor	kBench return a va	alue other than 1096
	in 'daysdiff' column		
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value		
	3 DateFetch script function get_c	**	es
	4 DateDim table is not setup properly		
	5 Database is not setup correctly		I
Priority	High	Priority Reason	Past 3 years date
			values are
Dui autha Canana	11.47		required
Priority Sequence	H-17		

ID	TC_27.0		
Test Case Name	No duplicate dates or tuples in datedim table		
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019
Description	No duplicate dates or tuples in datedir	n table	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Us</b>	ername = root Pa	ssword = password
	2 Data table exists		
	3 All the fields in the table contain va	lues	
Postconditions	1 Tables stay intact in the database		
-	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_27.0.sql	it. Fatar kay an a Di	_
Eveneted Decult	4 Execute the script or press Ctrl+Shif	•	
Expected Result	'cnt' column must return no value in N record returned.	nysql workbench.	mere snould be no
Test Entry/Exit		during testing phas	se or later when the
Criteria	<b>Entry:</b> Test can be started at any time during testing phase or later when the application is deployed. Test itself has no direct impact on the working of the		
Circiia	application.		
	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database. Extra or less values could cause issues that will		
	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	value or record
	Failed: The Result Grid in MySQL Wor	rkBench return a va	lue or record
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value		
	3 DateFetch script function get_c	**	25
	4 DateDim table is not setup pro	•	
- · · ·	5 Database is not setup correctly		
Priority	High	Priority Reason	Duplicate values
			can lead to inflated
			calculations and
			affect front-end
Priority Sequence	H-18		anect nont-end
Thority Sequence	11-10		

ID	TC_28.0		
Test Case Name	In Statisticalreturns table, cash on hand value must be >=0 for all records		
Created By	Abdul Ahmad Date Created 11-23-2019		
Description	Statisticalreturns table, cash on hand v	alue must be >=0 f	or all records
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	ired	
Executed By	Application User		
Preconditions	<ul> <li>Database is created with MySQL Username = root Password = password</li> <li>Data table exists</li> <li>All the fields in the table contain values</li> </ul>		
Postconditions	<ol> <li>Tables stay intact in the databa</li> </ol>	se	
	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_28.0.sql		
	4 Execute the script or press Ctrl+Shift+Enter key on a Windows		
Formando d Danvilla	machine		
Expected Result	No value should be returned in the 'cashonhand' column in MySQL		
Test Entry/Exit	WorkBench. There should be no record returned.		
Criteria	<b>Entry:</b> Test can be started at any time during testing phase or later when the application is deployed. Test itself has no direct impact on the working of the		
Circiia	application.		
	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database. Extra or	• •	
	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	value or record
	Failed: The Result Grid in MySQL Wor	kBench return a va	lue or record
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value		e table
	3 TradingSimulator.py had issues		
	4 Database is not setup correctly		
Priority	Medium	Priority Reason	Cash On Hand is
			used behind the
			scenes in the
Driority Convence	M 6		trading portfolio
Priority Sequence	M-6		

ID	TC_29.0			
Test Case Name	In Statisticalreturns table, 'portfoliovalue' data values must be >= 'cash on			
	hand' values for all records			
Created By	Abdul Ahmad Date Created 11-23-2019			
Description	In Statisticalreturns table, 'portfolioval	lue' data values mu	st be >= 'cash on	
	hand' values for all records			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ıired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	assword = password	
	2 Data table exists and all the fields	in the table contain	values	
Postconditions	1 Tables stay intact in the database			
	2 Database stays intact			
Test Steps	1 Open MySQL WorkBench			
	2 Browse to Unit Test folder			
	3 Open TC_29.0.sql			
	4 Execute the script or press Ctrl+Sh	ift+Enter key on a \	Windows machine	
Expected Result	No value should be returned in the My	/SQL WorkBench. T	here should be no	
	record returned.			
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the			
Criteria	application is deployed. Test itself has no direct impact on the system.			
	Exit: Test can be exited at any time during the run or after it finishes.			
Test Risks	This test itself causes no risk to the operation of the application or the			
	existing data in the database. Extra or less values could cause issues that will			
	need to be corrected by a developer.			
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo			
	Failed: The Result Grid in MySQL Wor		alue or record	
Fail Reasons	1 DataMain did not run properly			
	2 Python script did not pipe value		e table	
	3 TradingSimulator.py had issues			
	4 Database is not setup correctly		T	
Priority	Medium	Priority Reason	'portfoliovalue' is	
			always >= 'cash	
			on hand' because	
			it's the total of	
			starter cash +	
			(gain – loss)	
<b>Priority Sequence</b>	M-7			

ID	TC_30.0			
Test Case Name	In Statisticalreturns table, 'positionsize' column must be an integer data type			
Created By	Abdul Ahmad	Date Created	11-23-2019	
Description	In Statisticalreturns table, 'positionsize' column must be an integer data type			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	iired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Us</b>	ername = root Pa	ssword = password	
	2 Data table exists	2 Data table exists		
	3 All the fields in the table contain va	lues		
Postconditions	1 Tables stay intact in the database			
	2 Database stays intact			
Test Steps	1 Open MySQL WorkBench			
	2 Browse to Unit Test folder			
	3 Open TC_30.0.sql			
	4 Execute the script or press Ctrl+Shif			
Expected Result	'DATA_TYPE' column must show a valu	•		
Test Entry/Exit	Entry: Test can be started at any time			
Criteria	application is deployed. Test itself has no direct impact on the working of the			
	application.			
Test Risks	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.			
Test Nisks	This test itself causes no risk to the operation of the application or the existing data in the database. Extra or less values could cause issues that will			
	need to be corrected by a developer.			
Pass/Fail Criteria	Passed: The Result Grid in MySQL WorkBench returns single record and			
r assyr an enteria	column 'DATA TYPE' contains a value		igie record and	
	Failed: The Result Grid in MySQL Wor		TA TYPE' value	
	other than 'int'			
Fail Reasons	1 DataMain did not run properly			
	2 Python script did not pipe value	es properly into the	table	
	3 TradingSimulator.py had issues			
	4 Database is not setup correctly			
Priority	Medium	<b>Priority Reason</b>	Shares are not	
			bought or sold in	
			fractions, position	
			size represents	
			number of shares	
Priority Sequence	M-8			

ID	TC_31.0		
Test Case Name	Statisticalreturns table must contain 7 strategies		
Created By	Abdul Ahmad	Date Created	11-23-2019
Description	Statisticalreturns table must contain 7 strategies. These are strategy codes		
	use for different buysell trading simulations.		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	assword = password
	2 Data table exists		
	3 All the fields in the table contain va	alues	
Postconditions	1 Tables stay intact in the database		
_	2 Database stays intact		
Test Steps	1 Open MySQL WorkBench		
	2 Browse to Unit Test folder		
	3 Open TC_31.0.sql	Str. Fataulian an a N	A/: a al accesa and a la ima
Evenetad Decult	4 Execute the script or press Ctrl+Shi	•	
Expected Result Test Entry/Exit	'stratcode' column must show a value	•	
Criteria	<b>Entry:</b> Test can be started at any time during testing phase or later when the		
Criteria	application is deployed. Test itself has no direct impact on the working of the application.		
	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database. Extra or less values could cause issues that will		
	need to be corrected by a developer.		
Pass/Fail Criteria	Passed: The Result Grid in MySQL WorkBench returns single record and		
	column 'stratcode' contains a value of 7		
	Failed: The Result Grid in MySQL Wor	kBench shows 'stra	atcode' value other
	than 7		
Fail Reasons	1 DataMain did not run properly		
	2 Python script did not pipe value	es properly into the	table
	3 TradingSimulator.py had issues		
	4 Database is not setup correctly		
Priority	High	Priority Reason	All strategies
			should be
			computed
Priority Sequence	H-19		

ID	TC_32.0					
Test Case Name	Fibonacci lines values are in order, 'highfrllinelong' values are >=					
	'lowfrllinelong' values for all records					
Created By	Abdul Ahmad Date Created 11-23-2019					
Description	Fibonacci lines values are in order, 'hig	hfrllinelong' values	s are >=			
	'lowfrllinelong' values for all records					
Tester	Data Analyst or Developer					
Test Frequency	Anytime the application testing is requ	iired				
Executed By	Application User					
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	<b>assword</b> = password			
	2 Data table exists					
	3 All the fields in the table contain va	alues				
Postconditions	1 Tables stay intact in the database					
	2 Database stays intact					
Test Steps	1 Open MySQL WorkBench					
	2 Browse to Unit Test folder					
	3 Open TC_32.0.sql	···				
E control Book In	4 Execute the script or press Ctrl+Sh	•				
Expected Result	No records should be returned in the N	· ·				
Test Entry/Exit Criteria	<b>Entry:</b> Test can be started at any time					
Criteria	application is deployed. Test itself has	no direct impact of	n the working of the			
	application.					
Test Risks	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.  This test itself causes no risk to the operation of the application or the					
rese misks	existing data in the database. Extra or					
	need to be corrected by a developer.	iess values could et	dase issues that will			
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	results			
· ass, · an enterior	Failed: The Result Grid in MySQL Wor					
Fail Reasons	1 DataMain did not run properly					
	2 Python script did not pipe value	es properly into the	e table			
	3 EngineeredFeatures.py had issi					
	4 Database is not setup correctly					
Priority	Medium	<b>Priority Reason</b>	FRL strategy and			
			technical			
			indicators view			
			will be affected			
<b>Priority Sequence</b>	M-9					

ID	TC_33.0				
Test Case Name	Bollinger Bands values are in order, boll_ub_v > boll_lb_v values for all				
	records				
Created By	Abdul Ahmad Date Created 11-23-2019				
Description	Bollinger Bands values are in order, boll_ub_v > boll_lb_v values for all				
	records				
Tester	Data Analyst or Developer				
Test Frequency	Anytime the application testing is requ	iired			
Executed By	Application User				
Preconditions	1 Database is created with MySO	L <b>Username</b> = root	Password =		
	password				
	2 Data table exists				
	3 All the fields in the table contai				
Postconditions	3 Tables stay intact in the databa	ise			
	Database stays intact				
Test Steps	5 Open MySQL WorkBench				
	6 Browse to Unit Test folder				
	7 Open TC_33.0.sql	. CL:ft . E	va.c. I		
	8 Execute the script or press Ctrl+Shift+Enter key on a Windows				
Formanda di Dancolti	machine				
Expected Result	No records should be returned in the MySQL WorkBench.				
Test Entry/Exit Criteria	1	<b>Entry :</b> Test can be started at any time during testing phase or later when the			
Criteria	application is deployed. Test itself has no direct impact on the working of the				
	application.  Evit: Test can be evited at any time during the run or after it finishes				
Test Risks	<b>Exit</b> : Test can be exited at any time during the run or after it finishes.  This test itself causes no risk to the operation of the application or the				
Test Misks	existing data in the database. Extra or	• •			
	need to be corrected by a developer.	icss values could co	dusc issues that will		
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	results		
· ass, · an enterior	Failed: The Result Grid in MySQL Wor				
Fail Reasons	1 DataMain did not run properly				
	2 Python script did not pipe value	es properly into the	e table		
	3 EngineeredFeatures.py had issues				
	4 Database is not setup correctly				
Priority	Medium	<b>Priority Reason</b>	Technical		
			Indicators view		
			will be impacted		
<b>Priority Sequence</b>	M-10				

ID	TC_34.0					
Test Case Name	Each date must have a 0, 1, -1 signal for each strategy and symbol in					
	'actionsignals' table					
Created By	Abdul Ahmad Date Created 11-23-2019					
Description	Each date must have a 0, 1, -1 signal for	or each strategy an	nd symbol in			
	'actionsignals' table					
Tester	Data Analyst or Developer					
Test Frequency	Anytime the application testing is requ	iired				
Executed By	Application User					
Preconditions	1 Database is created with MySQL <b>U</b>	sername = root Pa	assword = password			
	2 Data table exists					
	3 All the fields in the table contain va					
Postconditions	1 Tables stay intact in the databa	se				
	2 Database stays intact					
Test Steps	1 Open MySQL WorkBench					
	2 Browse to Unit Test folder					
	3 Open TC_34.0.sql	. Cl.:Cl.:C	<b>D</b> C			
	4 Execute the script or press Ctrl-	•				
Expected Result	'cnt' values for 'SubTotal of Records Gi					
	equal to 'Total Records in 'actionsignals' Table row in the MySQL WorkBench.					
Test Entry/Exit	Entry: Test can be started at any time	during tosting pha	co or later when the			
Criteria	application is deployed. Test itself has					
Criteria		•	• •			
Test Risks	<b>Exit</b> : Test can be exited at any time during the run or after it finishes.  This test itself causes no risk to the operation of the application or the					
rest rusks	existing data in the database. Extra or					
	need to be corrected by a developer.	icas values coula co	adse issues that will			
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns 5	records with last			
	two records showing same value for 'c					
	Failed: The Result Grid in MySQL Wor		ferent values for			
	last two rows in 'cnt' column					
Fail Reasons	1 DataMain did not run properly					
	2 Python script did not pipe value	es properly into the	e table			
	3 BuySell.py had issues					
	4 Database is not setup correctly					
Priority	High	<b>Priority Reason</b>	BuySell signal to			
			indicate Buy(1),			
			Sell(-1) or Hold(0)			
Priority Sequence	H-20					

ID	TC_35.0				
Test Case Name	In Statisticalreturns table 'cash on hand' values must be between 0 and				
	'portfoliovalue', it should never exceed the 'portfolio' value				
Created By	Abdul Ahmad Date Created 11-23-2019				
Description	In Statisticalreturns table 'cash on har	nd' values must be k	etween 0 and		
	'portfoliovalue', it should never exceed the 'portfolio' value				
Tester	Data Analyst or Developer				
Test Frequency	Anytime the application testing is requ	uired			
Executed By	Application User				
Preconditions	1 Database is created with MySQL <b>U</b>	<b>Jsername</b> = root <b>P</b>	<b>assword</b> = password		
	2 Data table exists				
	3 All the fields in the table contain v				
Postconditions	1 Tables stay intact in the databa	ase			
	2 Database stays intact				
Test Steps	1 Open MySQL WorkBench				
	2 Browse to Unit Test folder				
	3 Open TC_35.0.sql	L Chift   Entar kay a	a a DC		
Expected Result	4 Execute the script or press Ctrl	•			
Expected Result	'mincoh' value must be >0 and 'maxcoh' must be < 'maxportfolio' in the MySQL WorkBench.				
Test Entry/Exit	Entry: Test can be started at any time during testing phase or later when the				
Criteria	application is deployed. Test itself has no direct impact on the system.				
	Exit: Test can be exited at any time di	•	•		
Test Risks	This test itself causes no risk to the operation of the application or the				
	existing data in the database. Extra or less values could cause issues that will				
	need to be corrected by a developer.				
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	orkBench returns si	ngle record with		
	minimum 'cashonhand' > 0 and maxin	num 'cashonhand' i	must be less than		
	'maxportfolio' value.				
	Failed: The Result Grid in MySQL Wo		lues that are greater		
	than 'maxportfolio' compared to 'max				
Fail Reasons	1 DataMain did not run properly				
	2 Python script did not pipe valu		e table		
	3 TradingSimulator.py had issues				
Driority	4 Database is not setup correctly		Important for		
Priority	High	Priority Reason	Important for buy/selling trades		
Driority Sequence	H-21		Duy/selling traues		
Priority Sequence	П-71				

ID	TC_36.0					
Test Case Name	BuyHold strategy must have no signals, this strategy should only hold the					
	stock and not sell it or buy more					
Created By	Aalem Singh Date Created 11-23-2019					
Description	BuyHold strategy must have no signals	, this strategy shou	ıld only hold the			
	stock and not sell it or buy more					
Tester	Data Analyst or Developer					
Test Frequency	Anytime the application testing is requ	iired				
Executed By	Application User					
Preconditions	<ol> <li>Database is created with MySQ</li> </ol>	L <b>Username</b> = root	Password =			
	password					
	2 Data table exists					
	3 All the fields in the table contai	n values				
Postconditions	<ol> <li>Tables stay intact in the databa</li> </ol>	se				
	2 Database stays intact					
Test Steps	1 Open MySQL WorkBench					
	2 Browse to Unit Test folder					
	3 Open TC_36.0.sql					
	4 Execute the script or press Ctrl-					
Expected Result	No value or record should be returned	•				
Test Entry/Exit	Entry: Test can be started at any time					
Criteria	application is deployed. Test itself has no direct impact on the working of the					
	application.					
	<b>Exit</b> : Test can be exited at any time during the run or after it finishes.					
Test Risks	This test itself causes no risk to the ope	•				
	existing data in the database. Extra or	less values could ca	ause issues that will			
December 1 Construction	need to be corrected by a developer.	d Danielanda and an an	-11			
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo		· · · · · · · · · · · · · · · · · · ·			
Fail Daggers	Failed: The Result Grid in MySQL Wor	rkbench returns a v	raiue or a record			
Fail Reasons	1 DataMain did not run properly	as arabarly into the	+abla			
	2 Python script did not pipe value	es properly into the	e table			
	3 BuySell.py had issues 4 Database is not setup correctly					
Priority	Medium	<b>Priority Reason</b>	BuyHold strategy			
THOTILY	Medialli	Triority Reason	is not supposed to			
			do trades			
Priority Sequence	M-11		ao tidacs			
. Hority ocquerice	171 44					

ID	TC_37.0					
Test Case Name	In 'actionsignals' table frl, cma, ema, macd signals sum must be between -5					
	and 5					
Created By	Aalem Singh Date Created 11-23-2019					
Description	In 'actionsignals' table frl, cma, ema, m	nacd signals sum m	ust be between -5			
	and 5					
Tester	Data Analyst or Developer					
Test Frequency	Anytime the application testing is requ	iired				
Executed By	Application User					
Preconditions	1 Database is created with MySQL <b>Us</b>	ername = root Pa	ssword = password			
	2 Data table exists	_				
	3 All the fields in the table contain va	lues				
Postconditions	1 Tables stay intact in the database					
	2 Database stays intact					
Test Steps	1 Open MySQL WorkBench					
	2 Browse to Unit Test folder					
	3 Open TC_37.0.sql	N. F. I	•			
E consideration	4 Execute the script or press Ctrl+Shif	•				
Expected Result	No value or record should be returned in the MySQL WorkBench.					
Test Entry/Exit	<b>Entry :</b> Test can be started at any time during testing phase or later when the					
Criteria	application is deployed. Test itself has no direct impact on the working of the application.					
	<b>Exit:</b> Test can be exited at any time during the run or after it finishes.					
Test Risks	This test itself causes no risk to the operation of the application or the					
i est nisks	existing data in the database. Extra or	• •				
	need to be corrected by a developer.	icss values could co	dusc issues that will			
Pass/Fail Criteria	Passed: The Result Grid in MySQL Wo	rkBench returns no	values/records			
. addy rain efficient	Failed: The Result Grid in MySQL Wor		•			
Fail Reasons	1 DataMain did not run properly	2				
	2 Python script did not pipe value	es properly into the	e table			
	3 BuySell.py had issues	, , ,				
	4 Database is not setup correctly	,				
Priority	Medium	<b>Priority Reason</b>	These strategies			
			signals are			
			combined into our			
			custom algorithm			
			for trading			
<b>Priority Sequence</b>	M-12					

ID	TC_38.0				
Test Case Name	getdatasources() function returns correct number of symbols				
Created By	John Gettel Date Created 11/24/2019				
Description	getdatasources() function returns of	correct number of	symbols.		
Tester	Data Analyst or Developer				
Test Frequency	During the testing phase or anytime	e in the future, the	re are n	o restrictions	
Executed By	Application User, Developer or any	one performing Q	۹ Testing	5	
Preconditions	<ol> <li>PyCharm is open</li> <li>Project is open in PyCharm</li> <li>MySQL is working properly</li> </ol>				
Postconditions	<ol> <li>Test is completed successfu</li> <li>Middle layer of Python will</li> <li>Database will be loaded wit</li> </ol>	push data to the N	-	tabase properly	
Test Steps	<ul> <li>1 Open PyCharm</li> <li>2 Navigate to Fistertab → F2019 → Unit Tests</li> <li>3 Open TC_38.0.py</li> <li>4 Click on Run menu → Run button</li> <li>5 The test is executed</li> </ul>				
Expected Result	Tester should receive the following or no message at all	message: "incorre	ect numb	per of symbols:"	
Test Entry/Exit Criteria	<b>Entry:</b> Test can be executed at any <b>Exit:</b> When the test completes or it	•	d during	the run	
Test Risks	No risk to the operation of the application or the existing data in the database.				
Pass/Fail Criteria	Passed: If no message Failed: If the message is "incorrect number of symbols:"				
Fail Reasons	4 No symbols entered into database 5 Incorrect symbols entered into database				
Priority	High	Priority Reason		orrect symbols dication to run ly	
<b>Priority Sequence</b>	H-22				

ID	TC_39.0			
Test Case Name	getdatasources() function returns correct ticker symbols			
Created By	John Gettel Date Created 11/24/2019			
Description	getdatasources() function returns (	correct ticker symb	ols.	
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytim			
Executed By	Application User, Developer or any	one performing Q	4 Testin	3
Preconditions	<ol> <li>PyCharm is open</li> <li>Project is open in PyCharm</li> <li>MySQL is working properly</li> </ol>			
Postconditions	<ol> <li>Test is completed successful</li> <li>Middle layer of Python will</li> <li>Database will be loaded wit</li> </ol>	, push data to the M	•	itabase properly
Test Steps	<ul> <li>1 Open PyCharm</li> <li>2 Navigate to Fistertab → F2019 → Unit Tests</li> <li>3 Open TC_39.0.py</li> <li>4 Click on Run menu → Run button</li> <li>5 The test is executed</li> </ul>			
Expected Result	Tester should receive the following message: "missing ticker:" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any	time in any order		
Criteria	<b>Exit:</b> When the test completes or it	can be terminated	d during	the run
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message Failed: If the message is "missing ticker:"			
Fail Reasons	1 Incorrect symbols entered i		1	
Priority	High	Priority Reason		correct symbols plication to run ly
<b>Priority Sequence</b>	H-23			

ID	TC_40.0				
Test Case Name	get_data () function runs successfully				
Created By	John Gettel Date Created 11/24/2019				
Description	get_data () function runs successfully				
Tester	Data Analyst or Developer				
Test Frequency	During the testing phase or anytime in the future, there are no restrictions				
Executed By	Application User, Developer or anyone performing QA Testing				
Preconditions	<ul> <li>1 PyCharm is open</li> <li>2 Project is open in PyCharm</li> <li>3 MySQL is working properly</li> <li>4 Internet connection is working properly</li> </ul>				
Postconditions	<ol> <li>Test is completed successfully</li> <li>Data is retrieved from data source</li> <li>Middle layer of Python will push data to the MySQL database</li> <li>Database will be loaded with accurate records</li> </ol>				
Test Steps	<ul> <li>1 Open PyCharm</li> <li>2 Navigate to Fistertab → F2019 → Unit Tests</li> <li>3 Open TC_40.0.py</li> <li>4 Click on Run menu → Run button</li> <li>5 The test is executed</li> </ul>				
Expected Result	Tester should receive the following message: "Incorrect return value while running get_data():" or no message at all				
Test Entry/Exit	Entry: Test can be executed at any time in any order				
Criteria	Exit: When the test completes or it can be terminated during the run				
Test Risks	No risk to the operation of the application or the existing data in the database.				
Pass/Fail Criteria	Passed: If no message Failed: If the message is "Incorrect return value while running get data():"				
Fail Reasons	MySQL database is not function properly     Internet connection not functioning properly				
Priority	High Priority Reason Need to retrieve data from external data source				
Priority Sequence	H-24				

ID	TC_41.0		
Test Case Name	get_calendar() function runs successfully		
Created By	John Gettel Date Created 11/24/2019		
Description	get_calendar() function runs successfully		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in the future, there are no restrictions		
Executed By	Application User, Developer or anyone performing QA Testing		
Preconditions	1 PyCharm is open		
	2 Project is open in PyCharm		
	3 MySQL is working properly		
	4 Internet connection is working properly		
Postconditions	1 Test is completed successfully		
	2 Data is retrieved from data source		
	3 Middle layer of Python will push data to the MySQL database		
	4 Database will be loaded with accurate records		
Test Steps	1 Open PyCharm		
	2 Navigate to Fistertab → F2019 → Unit Tests		
	3 Open TC_41.0.py		
	4 Click on Run menu → Run button		
	5 The test is executed		
Expected Result	Tester should receive the following message: "Incorrect return value while		
	running get_calendar():" or no message at all		
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	Exit: When the test completes or it can be terminated during the run		
Test Risks	No risk to the operation of the application or the existing data in the		
	database.		
Pass/Fail Criteria	Passed: If no message		
	Failed: If the message is "Incorrect return value while running		
	get_calendar():"		
Fail Reasons	MySQL database is not function properly		
	2 Internet connection not functioning properly		
Priority	Medium Priority Reason Need to retrieve		
	accurate dates to map		
	trading calendar		
<b>Priority Sequence</b>	M-13		

ID	TC_42.0				
Test Case Name	calculate() function runs successful	ly			
Created By	John Gettel Date Created 11/24/2019				
Description	calculate() function runs successful	ly			
Tester	Data Analyst or Developer				
Test Frequency	During the testing phase or anytim	e in the future, the	ere are n	o restrictions	
Executed By	Application User, Developer or any	one performing Q	A Testing	5	
Preconditions	<ol> <li>PyCharm is open</li> <li>Project is open in PyCharm</li> <li>MySQL is working properly</li> </ol>				
Postconditions	<ol> <li>Test is completed successful</li> <li>Middle layer of Python will</li> <li>Database will be loaded with</li> </ol>	push data to the N	-	tabase	
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F2019 → Unit Tests  3 Open TC_42.0.py  4 Click on Run menu → Run button  5 The test is executed				
Expected Result	Tester should receive the following message: "Incorrect return value while running calculate():" or no message at all				
Test Entry/Exit	Entry: Test can be executed at any time in any order				
Criteria	<b>Exit:</b> When the test completes or it can be terminated during the run				
Test Risks	No risk to the operation of the application or the existing data in the database.				
Pass/Fail Criteria	Passed: If no message				
	Failed: If the message is "Incorrect		running	calculate():"	
Fail Reasons	1 MySQL database is not fund				
Priority	High	Priority Reason	indicat	o add technical or calculations base for later tions	
<b>Priority Sequence</b>	H-25				

ID	TC_43.0			
Test Case Name	calculate_forecast() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	calculate_forecast() function runs	successfully		
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytim	e in the future, the	ere are n	o restrictions
Executed By	Application User, Developer or any	one performing Q	A Testin	3
Preconditions	<ol> <li>PyCharm is open</li> <li>Project is open in PyCharm</li> <li>MySQL is working properly</li> </ol>			
Postconditions	Test is completed successfu     Middle layer of Python will     Database will be loaded with	push data to the N	-	atabase
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F2019 → Unit Tests  3 Open TC_43.0.py  4 Click on Run menu → Run button  5 The test is executed			
Expected Result	Tester should receive the following message: "Incorrect return value while running calculate_forecast():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message Failed: If the message is "Incorrect return value while running calculate_forecast():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium	Priority Reason		ation for future orecast
<b>Priority Sequence</b>	M-14			

ID	TC_44.0			
Test Case Name	calculate_arima_forecast() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	calculate_arima_forecast() functio	n runs successfully		
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytim	e in the future, the	ere are n	o restrictions
Executed By	Application User, Developer or any	one performing Q	A Testing	3
Preconditions	<ol> <li>PyCharm is open</li> <li>Project is open in PyCharm</li> </ol>	, 1		
	3 MySQL is working properly			
Postconditions	1 Test is completed successfu	ılly		
	2 Middle layer of Python will	push data to the N	1ySQL da	ntabase
	3 Database will be loaded wit	th accurate records	5	
Test Steps	1 Open PyCharm			
	2 Navigate to Fistertab → F20	019 → Unit Tests		
	3 Open TC_44.0.py			
	4 Click on Run menu → Run button			
	5 The test is executed			
Expected Result	Tester should receive the following	g message: "Incorre	ect retur	n value while
	running calculate_arima_forecast():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any	time in any order		
Criteria	Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the			
	database.			
Pass/Fail Criteria	Passed: If no message			
	Failed: If the message is "Incorrect return value while running			
	calculate_arima_forecast():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium	Priority Reason	Calcula forecas	ntion for arima
<b>Priority Sequence</b>	M-15			

ID	TC_45.0			
Test Case Name	calculate_random_forest_forecast() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	calculate_random_forest_forecast() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restrictions			
Executed By	Application User, Developer or anyone performing QA Testing			
Preconditions	1 PyCharm is open			
	2 Project is open in PyCharm			
	3 MySQL is working properly			
Postconditions	1 Test is completed successfully			
	2 Middle layer of Python will push data to the MySQL database			
	3 Database will be loaded with accurate records			
Test Steps	Following are the steps			
	1 Open PyCharm			
	2 Navigate to Fistertab → F2019 → Unit Tests			
	3 Open TC_45.0.py			
	4 Click on Run menu → Run button			
	5 The test is executed			
<b>Expected Result</b>	Tester should receive the following message: "Incorrect return value while			
	running calculate_random_forest_forecast():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the			
	database.			
- /- !! o !: !				
Pass/Fail Criteria	Passed: If no message			
	Failed: If the message is "Incorrect return value while running			
Foil December	calculate_random_forest_forecast():"			
Fail Reasons	1 MySQL database is not function properly			
Priority	Medium Priority Reason Calculation for random forest forecast			
Priority Sequence	M-16			
Thority Sequence	IALTO			

ID	TC_46.0			
Test Case Name	calculate_svm_forecast() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	calculate_svm_forecast() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytim	e in the future, the	re are n	o restrictions
Executed By	Application User, Developer or any	one performing Q	۹ Testin	3
Preconditions	<ul><li>1 PyCharm is open</li><li>2 Project is open in PyCharm</li><li>3 MySQL is working properly</li></ul>			
Postconditions	<ol> <li>Test is completed successfully</li> <li>Middle layer of Python will push data to the MySQL database</li> <li>Database will be loaded with accurate records</li> </ol>			
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F20  3 Open TC_46.0.py  4 Click on Run menu → Run b  5 The test is executed			
Expected Result	Tester should receive the following message: "Incorrect return value while running calculate_svm_forecast():" or no message at all			
Test Entry/Exit Criteria	Entry: Test can be executed at any time in any order			
Criteria	<b>Exit:</b> When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message Failed: If the message is "Incorrect return value while running calculate_svm_forecast():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium	Priority Reason		ntion for support machine st
<b>Priority Sequence</b>	M-17			

ID	TC_47.0		
Test Case Name	calculate_ forecast_old() function runs successfully		
Created By	John Gettel Date Created 11/24/2019		
Description	calculate_forecast_old() function runs successfully		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in the future, there are no restrictions		
Executed By	Application User, Developer or anyone performing QA Testing		
Preconditions	1 PyCharm is open		
	2 Project is open in PyCharm		
	3 MySQL is working properly		
Postconditions	1 Test is completed successfully		
	2 Middle layer of Python will push data to the MySQL database		
	3 Database will be loaded with accurate records		
Test Steps	Following are the steps		
	1 Open PyCharm		
	2 Navigate to Fistertab → F2019 → Unit Tests		
	3 Open TC_47.0.py		
	4 Click on Run menu → Run button		
	5 The test is executed		
Expected Result	Tester should receive the following message: "Incorrect return value while		
	running calculate_forecast_old():" or no message at all		
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	Exit: When the test completes or it can be terminated during the run		
Test Risks	No risk to the operation of the application or the existing data in the		
	database.		
- /- !! - !			
Pass/Fail Criteria	Passed: If no message		
	<b>Failed:</b> If the message is "Incorrect return value while running calculate_		
Fail Pagents	forecast_old():"		
Fail Reasons	1 MySQL database is not function properly  Medium Priority Reason Calculation for old		
Priority	Medium Priority Reason Calculation for old forecast model		
Priority Sequence	M-18		
. Hority Sequence	10		

ID	TC_48.0			
Test Case Name	calculate_xgboost_forecast() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	calculate_xgboost_forecast() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytim	e in the future, the	re are n	o restrictions
Executed By	Application User, Developer or any	one performing Q	۹ Testin	3
Preconditions	<ul><li>1 PyCharm is open</li><li>2 Project is open in PyCharm</li><li>3 MySQL is working properly</li></ul>			
Postconditions	<ol> <li>Test is completed successful</li> <li>Middle layer of Python will</li> <li>Database will be loaded wit</li> </ol>	push data to the N	•	atabase
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F2C  3 Open TC_48.0.py  4 Click on Run menu → Run b  5 The test is executed			
<b>Expected Result</b>	Tester should receive the following message: "Incorrect return value while running calculate_xgboost_forecast():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	<b>Exit:</b> When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message Failed: If the message is "Incorrect return value while running calculate_xgboost_forecast():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium	Priority Reason		ntion for nt boost st
<b>Priority Sequence</b>	M-19			

ID	TC_49.0			
Test Case Name	cma_signal() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	cma_signal() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restrictions			
Executed By	Application User, Developer or anyone performing QA Testing			
Preconditions	<ul><li>1 PyCharm is open</li><li>2 Project is open in PyCharm</li></ul>			
	3 MySQL is working properly			
Postconditions	1 Test is completed successfully			
1 osteonarions	2 Middle layer of Python will push data to the MySQL database			
	3 Database will be loaded with accurate records			
Test Steps	Following are the steps			
	1 Open PyCharm			
	2 Navigate to Fistertab → F2019 → Unit Tests			
	3 Open TC_49.0.py			
	4 Click on Run menu → Run button			
	5 The test is executed			
<b>Expected Result</b>	Tester should receive the following message: "Incorrect return value while			
	running cma_signal():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the			
	database.			
Pass/Fail Criteria	Passed: If no message			
	Failed: If the message is "Incorrect return value while running cma_signal():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium Priority Reason CMA signal generation			
<b>Priority Sequence</b>	M-20			

ID	TC_50.0			
Test Case Name	frl_signal() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	frl_signal() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restrictions			
Executed By	Application User, Developer or anyone performing QA Testing			
Preconditions	1 PyCharm is open			
	2 Project is open in PyCharm			
	3 MySQL is working properly			
Postconditions	1 Test is completed successfully			
	Middle layer of Python will push data to the MySQL database			
	3 Database will be loaded with accurate records			
Test Steps	Following are the steps			
	1 Open PyCharm			
	2 Navigate to Fistertab → F2019 → Unit Tests			
	3 Open TC_50.0.py			
	4 Click on Run menu → Run button			
	5 The test is executed			
<b>Expected Result</b>	Tester should receive the following message: "Incorrect return value while			
	running frl_signal():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the			
1 CSC INISKS	database.			
	database.			
Pass/Fail Criteria	Passed: If no message			
	Failed: If the message is "Incorrect return value while running frl_signal():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium Priority Reason FRL signal generation			
<b>Priority Sequence</b>	M-21			

ID	TC_51.0			
Test Case Name	ema_signal() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	ema_signal() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restrictions			
Executed By	Application User, Developer or anyone performing QA Testing			
Preconditions	1 PyCharm is open			
	2 Project is open in PyCharm			
	3 MySQL is working properly			
Postconditions	1 Test is completed successfully			
	2 Middle layer of Python will push data to the MySQL database			
	3 Database will be loaded with accurate records			
Test Steps	Following are the steps			
	1 Open PyCharm			
	2 Navigate to Fistertab → F2019 → Unit Tests			
	3 Open TC_51.0.py			
	4 Click on Run menu → Run button			
	5 The test is executed			
Expected Result	Tester should receive the following message: "Incorrect return value while			
	running ema_signal():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the			
	database.			
Pass/Fail Criteria	Passad: If no mossage			
rass/raii Cillella	Passed: If no message Failed: If the message is "Incorrect return value while running ema_signal():"			
Fail Reasons	1 MySQL database is not function properly			
Priority	Medium Priority Reason EMA signal generation			
Priority Sequence	M-22			
The state of the s	···			

ID	TC_52.0		
<b>Test Case Name</b>	macd_signal() function runs successfully		
Created By	John Gettel Date Created 11/24/2019		
Description	macd_signal() function runs successfully		
Tester	Data Analyst or Developer		
Test Frequency	During the testing phase or anytime in the future, there are no restrictions		
Executed By	Application User, Developer or anyone performing QA Testing		
Preconditions	1 PyCharm is open		
	2 Project is open in PyCharm		
	3 MySQL is working properly		
Postconditions	1 Test is completed successfully		
	Middle layer of Python will push data to the MySQL database		
	3 Database will be loaded with accurate records		
Test Steps	Following are the steps		
	1 Open PyCharm		
	2 Navigate to Fistertab → F2019 → Unit Tests		
	3 Open TC_52.0.py		
	4 Click on Run menu → Run button		
	5 The test is executed		
<b>Expected Result</b>	Tester should receive the following message: "Incorrect return value while		
	running macd_signal():" or no message at all		
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	<b>Exit:</b> When the test completes or it can be terminated during the run		
Test Risks	No risk to the operation of the application or the existing data in the		
	database.		
Pass/Fail Criteria	Passed: If no message		
	Failed: If the message is "Incorrect return value while running macd_signal():"		
Fail Reasons	MySQL database is not function properly		
Priority	Medium Priority Reason Moving average		
	convergence/divergence		
	signal generation		
<b>Priority Sequence</b>	M-23		

ID	TC_53.0			
Test Case Name	algo_signal() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	algo_signal() function runs successi	<sup>F</sup> ully		
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime	e in the future, the	ere are n	o restrictions
Executed By	Application User, Developer or any	one performing Q	A Testing	3
Preconditions	<ul><li>1 PyCharm is open</li><li>2 Project is open in PyCharm</li><li>3 MySQL is working properly</li></ul>			
Postconditions	<ol> <li>Test is completed successfully</li> <li>Middle layer of Python will push data to the MySQL database</li> <li>Database will be loaded with accurate records</li> </ol>			
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F2019 → Unit Tests  3 Open TC_53.0.py  4 Click on Run menu → Run button  5 The test is executed			
Expected Result	Tester should receive the following message: "Incorrect return value while running algo_signal():" or no message at all			
Test Entry/Exit Criteria	Entry: Test can be executed at any time in any order  Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message	_		
Fail Dansaus	Failed: If the message is "Incorrect return value while running algo_signal():"			
Fail Reasons	1 MySQL database is not function properly			
Priority	Medium	Priority Reason	_	hm forecast- gnal generation
<b>Priority Sequence</b>	M-24			

ID	TC_54.0			
<b>Test Case Name</b>	trade_sim() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	trade_sim() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime in the future, there are no restricti	ions		
Executed By	Application User, Developer or anyone performing QA Testing			
Preconditions	<ul><li>1 PyCharm is open</li><li>2 Project is open in PyCharm</li><li>3 MySQL is working properly</li></ul>			
Postconditions	<ol> <li>Test is completed successfully</li> <li>Middle layer of Python will push data to the MySQL database</li> <li>Database will be loaded with accurate records</li> </ol>			
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F2019 → Unit Tests  3 Open TC_54.0.py  4 Click on Run menu → Run button  5 The test is executed			
Expected Result	Tester should receive the following message: "Incorrect return value while running trade_sim():" or no message at all			
Test Entry/Exit Criteria	Entry: Test can be executed at any time in any order Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message Failed: If the message is "Incorrect return value while running trade_sim():"			
Fail Reasons	MySQL database is not function properly			
Priority	Medium Priority Reason Individual trade strategy simulat			
<b>Priority Sequence</b>	M-25			

ID	TC_55.0			
Test Case Name	comb_sim() function runs successfully			
Created By	Abdul Ahmad; John Gettel; Date Created 11/24/2019			
Description	comb_sim() function runs successfully			
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anytime	e in the future, t	here are n	o restrictions
Executed By	Application User, Developer or any	one performing	QA Testin	g
Preconditions	<ul><li>1 PyCharm is open</li><li>2 Project is open in PyCharm</li><li>3 MySQL is working properly</li></ul>	2 Project is open in PyCharm		
Postconditions	<ul><li>1 Test is completed successfu</li><li>2 Middle layer of Python will</li></ul>	<ul><li>1 Test is completed successfully</li><li>2 Middle layer of Python will push data to the MySQL database</li></ul>		
Test Steps	Following are the steps  1 Open PyCharm  2 Navigate to Fistertab → F2019 → Unit Tests  3 Open TC_55.0.py  4 Click on Run menu → Run button  5 The test is executed			
Expected Result	Tester should receive the following message: "Incorrect return value while running comb_sim():" or no message at all			
Test Entry/Exit Criteria	Entry: Test can be executed at any time in any order  Exit: When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: If no message Failed: If the message is "Incorrect return value while running comb_sim():"			
Fail Reasons	1 MySQL database is not function properly			
Priority	Medium Priority Reason Combination trade			nation trade
		•		gy simulation
<b>Priority Sequence</b>	M-26			

ID	TC_56.0			
Test Case Name	buy_hold_sim() function runs successfully			
Created By	John Gettel Date Created 11/24/2019			
Description	buy_hold_sim() function runs succ	essfully		
Tester	Data Analyst or Developer			
Test Frequency	During the testing phase or anyting	e in the future, the	ere are n	o restrictions
Executed By	Application User, Developer or an	one performing Q	A Testing	3
Preconditions	1 PyCharm is open			
	<ol><li>Project is open in PyCharm</li></ol>			
	<ol><li>MySQL is working properly</li></ol>			
Postconditions	<ol> <li>Test is completed successf</li> </ol>	ılly		
	2 Middle layer of Python will		-	atabase
	3 Database will be loaded wi	th accurate records	5	
Test Steps	Following are the steps			
·	1 Open PyCharm			
	2 Navigate to Fistertab → F2019 → Unit Tests			
	3 Open TC 56.0.py			
	4 Click on Run menu → Run			
	5 The test is executed			
Expected Result	Tester should receive the followin	-	ect retur	n value while
	running buy_hold_sim ():" or no message at all			
Test Entry/Exit	Entry: Test can be executed at any	time in any order		
Criteria	<b>Exit:</b> When the test completes or it can be terminated during the run			
Test Risks	No risk to the operation of the app	lication or the exis	ting data	in the
	database.			
Pass/Fail Criteria	Passed: If no message			
1 433/1 dil Citteria	_	return value while	running	ī
	Failed: If the message is "Incorrect return value while running buy hold sim():"			
Fail Reasons	1 MySQL database is not function properly			
Priority	Medium Priority Reason Buy and hold trade			d hold trade
	strategy simulation			
<b>Priority Sequence</b>	M-27		_	

#### 3 NON-FUNCTIONAL TESTING

#### 3.1 NON-FUNCTIONAL TESTING APPROACH

Non-functional testing is done to verify the non-functional requirements of the application such as performance, usability, interface etc. All tests will be performed in Tableau. It is the front-end application platform to be used by the financial analyst/user. Tester will be performing all the test in Tableau while maintaining the connectivity to the locally installed MySQL database. Tester will have to setup a local instance of the MySQL database with **username** = root and **password** = password. This username and password could be different for each installation on each machine. In PyCharm 'dbEngine.py' will have to be configured accordingly. Testing should not begin before the aforementioned configurations are setup in each software.

#### 3.2 NON-FUNCTIONAL TESTING PASS/FAIL CRITERIA

Each test must produce the desired outcome or perform according to requirements to qualify for passed status. Testing is done in order to verify that in case of a failure the system is capable enough to keep functioning. If a test fails due to Tableau functionality issues, it is out of the scope of the project for the tester or developers to fix it as Tableau is a third-party proprietary software. Tester can notify the developers and client. They can report issue to Tableau vendor via email or online community forums.

#### 3.3 NON-FUNCTIONAL TESTING RISKS

There is no direct risk to any portions of the application during any non-functional tests. In our application these tests are all performed on Tableau which is a third-party software. All its functionality is out of the box. The only serious consequence could be that the application might close due to memory or other local hardware issues. There are no tests that insert or delete data from the database so all calculations will stay intact after a test is complete.

#### 3.4 NON-FUNCTIONAL TESTING EXPECTED RESULTS

Each test must perform according to the expectation otherwise the feature or function will be classified as failed.

#### 3.5 NON-FUNCTIONAL TESTING PRIORITY/PRIORITY REASON

Each test is prioritized according to the feature and use cases.

#### 3.6 NON-FUNCTIONAL TEST CASES

ID	NFTC_1.0			
Test Case Name	Tableau - Enter User Id and Password, configured locally for MySQL database			
	named 'gmfsp_db'			
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019	
Description	Tableau - Enter User Id and Password,	configured locally f	for MySQL database	
	named 'gmfsp_db'			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	iired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL User	rname = root Pass	word = password	
	2 Database and Tableau must be setup	properly		
	3 Required drivers are installed properl	<u> </u>		
Postconditions	1 Tableau will be connected to the da			
	2 Tableau will be able to pull all data	and show the upda	ted tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file name 'GM_FinTech_Application'			
	2 Double click to open it			
	You should see a window like the one on the right			
		'		
	5 Or enter any password setup during MySQL			
	database instance setup			
Expected Result	Tableau has established connection with the locally setup database			
Test Entry/Exit	Entry: Test can be executed at any time in any order			
Criteria	<b>Exit</b> : When the test completes, or it ca		_	
Test Risks	This test itself causes no risk to the op-	eration of the appli	cation or the	
	existing data in the database.			
Pass/Fail Criteria	Passed : Tableau is successfully connec	•	I DB 'gmfsp_db'	
	Failed: Tableau is not able to connec	t to the database		
Fail Reasons	MySQL database is not setup properly			
Priority	High	Priority Reason	Application will	
			not work without	
			a database	
			connection	
Priority Sequence	H-1			

ID	NFTC_2.0		
Test Case Name	Tableau - data is refreshed, check any one data-model		
Created By	Abdul Ahmad	Date Created	11-24-2019
Description	Tableau - data is refreshed, check any	one data-model	
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	<b>sword</b> = password
	2 Tableau must be setup properly		
	3 Database must be setup properly		
	4 Required drivers are installed proper	•	
Postconditions	1 Tableau will be connected to the d		
	2 Tableau will be able to pull all data	and show the upd	ated tabs
-	3 Application will be functional		
Test Steps	1 Browse to Tableau application file	name	ON MRE TARIENII
	'GM_FinTech_Application'	G DCPA_COMPARIS	ON TABLE***
	2 Double click to open it	0 FEATURES_IN_TAI	Edit Data Source  Refresh  Market Source
	3 Click on the <b>'Data Source'</b> tab to the very 8 FEATURES_IN_TABLEAU_FR View Data 9 FORECAST_IN_TABLEAU_FR View Data		
	left of the screen	O CODECACT IN TA	n r
	4 Right click on the data source with blue check at the bottom	U FORECAST_IN_TA	Class
	5 Press 'Refresh'	⊕ FORECAST_IN_TA	BLEAU_RA Close
Expected Result	Data Source is refreshed without any e	arror	
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	<b>Exit:</b> When the test completes, or it can	•	uring the run
Test Risks	This test itself causes no risk to the ope		
	existing data in the database.		
Pass/Fail Criteria	Passed: Tableau is successfully connec	cted to MySQL loca	I DB <b>'gmfsp_db'</b>
	Failed: Tableau is not able to connec	t to the database o	r refresh
Fail Reasons	1 MySQL database is not setup proper	ly	
	2 Tableau data source is not setup pro	perly, recreated the	e data source
Priority	High	<b>Priority Reason</b>	No related
			features will work
	without data		
			source refresh.
<b>Priority Sequence</b>	H-2		

ID	NFTC_3.0			
Test Case Name	Tableau - latest data for the base statistics is updated			
Created By	Abdul Ahmad Date Created 11-24-2019			
Description	Tableau - latest data for the base statistics is update	ated		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is required	Anytime the application testing is required		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Username</b> = ro	oot <b>Pas</b> s	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly			
	4 Required drivers are installed properly			
Postconditions	1 Tableau will be connected to the database			
	2 Tableau will be able to pull all data and show	the upda	ated tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file name		edictions (DCPA = 52.46)	
	'GM_FinTech_Application'	roved by excluding	g today's close , 10 day avg, 4 day momentur	
	2 Double click to open it	, Wh	<b>1 1 1 1 1 1 1 1 1 1</b>	
	3 Click on the tab named 'Predictions-			
	PrevGrp-XGB'			
	4 Hover over any Pink line graph to the point	1, 19 Nov 1, 19	Close, 33.33	
	on the most right as shown on the right	nt Boosting Pre	dictions (DCPA = 52.1/12)	
Expected Posult	5 A tooltip should pop up	okonds :	and halidays have	
Expected Result	Tooltip shows the most recent expected date, weekends and holidays have no data in this application.			
Test Entry/Exit	Entry: Test can be executed at any time in any or	rdor		
Criteria	· · ·		uring the run	
Test Risks	<b>Exit:</b> When the test completes, or it can be terminated during the run  This test itself causes no risk to the operation of the application or the			
rest rusks	existing data in the database.	лс аррп	cation of the	
Pass/Fail Criteria	Passed: Tableau is successfully connected to My	SOL loca	I DB 'gmfsp db'	
	and shows updated data			
	<b>Failed</b> : Tableau is not able to connect to the date	tabase o	r pull latest data	
Fail Reasons	MySQL database is not setup properly		<u> </u>	
	2 Tableau data source is not setup properly, recreated the data source			
	3 SQL Scripts for this tab has issues, multiple scripts			
	4 Python scripts related to this data did not run properly			
Priority	High Priority Reason Latest data and			
			forecasts requried	
<b>Priority Sequence</b>	Н-3			

ID	NFTC_4.0		
Test Case Name	Tableau - User can view all tabs		
Created By	Abdul Ahmad	Date Created	11-24-2019
Description	Tableau - User can view all tabs		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password
	2 Tableau must be setup properly		
	3 Database must be setup properly		
	4 Required drivers are installed proper	•	
Postconditions	1 Tableau will be connected to the d		
	2 Tableau will be able to pull all data	and show the upd	ated tabs
=	3 Application will be functional	(0)	
Test Steps	1 Browse to Tableau application file	name 'GM_FinTeci	n_Application'
	2 Double click to open it 3 Lock at the bottom of DCPA Comparison DCPA Comparison MAPE # Predictions-ARIMA-RFR-SVM # Predictions-PrevGrp-XGB /		
	5 Look at the bottom of		
	the application		
Expected Result	4 There should be tabs present with different names  Tabs available in the application, tester should be able to click on each tab.		
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	Exit: When the test completes, or it can be terminated during the run		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database.		
Pass/Fail Criteria	Passed: Tableau is successfully connec	cted to MySQL loca	I DB 'gmfsp db' and
	shows tabs without error	·	
	Failed: Tableau is not able to connec	t to the database a	nd show tabs
Fail Reasons	<ol> <li>MySQL database is not setup p</li> </ol>	roperly	
	3 Tableau data source is not setu	p properly, recreat	ed the data source
	2 Tableau has encountered an issue(we are limited in correcting		
	Tableau related technical issues as it's a third-party software)		
Priority	High	Priority Reason	Tabs house the
	graphs and		
	measures		
<b>Priority Sequence</b>	H-4		

ID	NFTC_5.0			
Test Case Name	Tableau - Toggle radio buttons are working, mostly available on the right			
Created By	Abdul Ahmad	Date Created	11-24-2019	
Description	Tableau - Toggle radio buttons are wor	king, mostly availa	ble on the right	
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly and	•	are installed	
Postconditions	1 Tableau will be connected to the d			
	2 Tableau will be able to pull all data	and show the upd	ated tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file	= 52.46)	instrumentname	
	name 'GM_FinTech_Application'	day avg, 4 day momentu	(All)  CARZ  GM	
	2 Double click to open it		- 38 Spring Spri	
		3 Click on the tab named 'Predictions-		
	•	PrevGrp-XGB'		
	4 On the right, radio buttons are visible Feb 1, 20 Mar 1, 20 Apr 1, 20 forecast doseprice			
	5 Make a different selection other than			
Eveneted Besult		the default selected option		
Expected Result	Toggle buttons are filtering data as per design. The graph should change			
Test Entry/Exit	when the tester toggles the button.  Entry: Test can be executed at any time	o in any order		
Criteria	<b>Exit:</b> When the test completes, or it can	•	uring the run	
Test Risks	This test itself causes no risk to the ope		_	
Test Misks	existing data in the database.	eration of the appli	ication of the	
Pass/Fail Criteria	Passed: Tableau is successfully connect	rted to MySOL loca	I DB 'gmfsn dh' and	
1 assyr an enteria	shows tabs without error	sted to iviyade loca	m bb giiiisp_ab ana	
	<b>Failed</b> : Tableau is not able to connect	t to the DB and do	es not show tabs	
Fail Reasons	MySQL database is not setup process.			
	2 Tableau data source is not setu		ted the data source	
	3 Tableau has encountered an iss	sue(we are limited	in correcting it)	
	4 SQL scripts to build this tab has issues, multiple scripts on this tab			
	5 Filters setting in Tableaus is not			
Priority	High	<b>Priority Reason</b>	Ability to view	
	different datasets			
			and graphs	
<b>Priority Sequence</b>	H-5			

ID	NFTC_6.0	
Test Case Name	Tableau - Zoom feature is working on all applicable line graphs	
Created By	Abdul Ahmad Date Created 11-24-2019	
Description	Tableau - Zoom feature is working on all applicable line graphs	
Tester	Data Analyst or Developer	
Test Frequency	Anytime the application testing is required	
Executed By	Application User	
Preconditions	1 Database is created with MySQL <b>Username</b> = root <b>Password</b> = password	
	2 Tableau must be setup properly	
	3 Database must be setup properly and required drivers are installed	
Postconditions	1 Tableau will be connected to the database	
	Tableau will be able to pull all data and show the updated tabs	
	3 Application will be functional	
Test Steps	1 Browse to Tableau application file name	
	'GM_FinTech_Application'	
	2 Double click to open it	
	3 Click on 'BuySellCMA' tab	
	4 Bring the cursor to the dark area	
	5 Using a touchpad if available use two fingers to	
	make a 200m in or 200m-out motion, similar to like	
	a silial controlle	
	6 Or right click in the dark area, a menu will	
	pop-up, click Show View Toolbar  7 A vertical bar like the one shown here should appear	
	7 A Vertical but like the one shown here should	
	· ·	
Expected Result	Zoomed in or out of the graph or object	
Test Entry/Exit	Entry: Test can be executed at any time in any order	
Criteria	<b>Exit:</b> When the test completes, or it can be terminated during the run	
Test Risks	This test itself causes no risk to the operation of the application.	
Pass/Fail Criteria	Passed: Tableau is successfully able to zoom-in or out of the graphs	
E-1 B	Failed: Tableau issue or data not loaded properly to create the object	
Fail Reasons	1 MySQL database is not setup properly	
	2 Tableau has encountered an issue(we are limited in correcting	
	Tableau related technical issues as it's a third-party software)	
	3 'BUY_SELL_SIGNALS_IN_TABLEAU_CMA.sql' script has issues	
Priority	4 'BuySell.py' Python script has issues  Medium Priority Reason zoom-in or out	
Priority Sequence	M-1	
Filolity Sequence	IAI-T	

ID	NFTC_7.0			
Test Case Name	Tableau - Forecast line graph shows future dates			
Created By	Abdul Ahmad	<b>Date Created</b>	11-24-2019	
Description	Tableau - Forecast line graph shows for	uture dates		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is req	uired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Us</b>	ername = root Pas	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly ar		are installed	
Postconditions	1 Tableau will be connected to the			
	2 Tableau will be able to pull all dat	a and show the upd	lated tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file	e name Moving Average F	orecast (DCPA = 56.0942)	
	'GM_FinTech_Application'			
	2 Double click to open it			
		3 Click on the 'ARIMA' tab		
	4 Hover the mouse pointer over the	e 'Green'	date: 12/2/2019 forecastcloseprice: 35.14	
	line where there is no 'Pink' line		1 111 111	
Expected Result	When the mouse hovers over the Green line future dates should be visible			
Test Entry/Exit	<b>Entry</b> : Test can be executed at any time in any order on any tab that shows predictions based on an algorithm			
Criteria		an ha tarminatad d	uring the run	
Test Risks	<b>Exit:</b> When the test completes, or it can be terminated during the run  This test itself causes no risk to the operation of the application or the			
i est risks	existing data in the database.	beration of the appr	ication of the	
Pass/Fail Criteria	Passed: Tableau is successfully connected to MySQL local DB 'gmfsp db' and			
1 assyrani ericeria	future forecast graph and data is conf	•	m bb giiiisp_ab ana	
	<b>Failed</b> : Tableau is not able to show the			
Fail Reasons	1 MySQL database is not setup	properly		
	2 Tableau data source is not set		ted the data source	
	3 Tableau has encountered an is	ssue(we are limited	in correcting	
	Tableau related technical issues as it's a third-party software)			
	4 'FORECAST_IN_TABLEAU_ARIMA_ONLY.sql' script has issues			
	5 Python script <b>'DataForecast.py'</b> has issues			
	6 'DateDim' table has issues in MySQL 'gmfsp_db' database			
Priority	Medium	<b>Priority Reason</b>	Forecast values	
			must be visible	
Priority Sequence	M-2			

ID	NFTC_8.0			
Test Case Name	Tableau - Date is always on x-axis on all graphs			
Created By	Abdul Ahmad	Date Created	11-24-2019	
Description	Tableau - Date is always on x-axis on a	ll graphs		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	iired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly and	•	are installed	
Postconditions	1 Tableau will be connected to the d			
	2 Tableau will be able to pull all data	and show the upd	ated tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file	name <b>'GM_FinTec</b> l	h_Application'	
	2 Double click to open it	28-		
	3 Click on 'BuySellCustom' tab			
	4 Look at the bottom of the graph	Nov 1, 18 Jan 1, 19 Mar :	l, 19 May 1, 19 Jul 1, 19 Sep 1, 19	
	5 The date values should be aligned			
	horizontally with the object			
Expected Result	Date values show on the horizontal x-axis			
Test Entry/Exit	<b>Entry :</b> Test can be executed at any time in any order on any tab that shows			
Criteria	date values			
	Exit: When the test completes, or it can be terminated during the run			
Test Risks	This test itself causes no risk to the ope	eration of the appli	cation or the	
Daniel (Fall Odina)	existing data in the database.	-1 - 1 - N4 COL 1	IDD ( (	
Pass/Fail Criteria	Passed: Tableau is successfully connected to MySQL local DB 'gmfsp_db' and			
	date values are displayed on the horizo		d aba data	
	<b>Failed</b> : Tableau is not able to connect	t to the database a	nd snow date	
Fail Reasons	values on the x-axis  1 MySQL database is not setup p	ronerly		
Tall Neasolls	2 Tableau data source is not setup p	• •	ed the data source	
	3 Tableau has encountered an iss			
		•	_	
		Tableau related technical issues as it's a third-party software)  4 'BUY_SELL_SIGNALS_IN_TABLEAU_ALGO.sql' script to has issues		
	5 <b>'BuySell.py'</b> Python script related to this tab has issues			
	6 <b>'DateDim'</b> table has issues in MySQL <b>'gmfsp_db'</b> database			
Priority	High	Priority Reason	Dates must be	
	-	•	shown on x-axis	
<b>Priority Sequence</b>	H-6			

ID	NFTC_9.0			
Test Case Name	Tableau - Close price is always on the y-axis on the left			
Created By	Abdul Ahmad Date Created 11-24-2019			
Description	Tableau - Close price is always on the	y-axis on the left		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is req	uired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Us</b>	ername = root Pas	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly ar	nd required drivers a	are installed	
Postconditions	1 Tableau will be connected to the	database		
	2 Tableau will be able to pull all dat	a and show the upd	ated tabs	
	3 Application will be functional			
Test Steps	<ol> <li>Browse to Tableau application</li> </ol>	file name 'GM_Fin'	Tech_Application'	
	2 Double click to open it	45-		
	3 Click on the <b>'FRLFeatures'</b> tab	40		
	4 Look for the 'Close' values on	the		
	left side y-axis of the graph			
<b>Expected Result</b>	'Close' values show on the left vertica	l y-axis		
Test Entry/Exit	Entry: Test can be executed at any time in any order on any tab showing			
Criteria	'Close' value			
	Exit: When the test completes, or it can be terminated during the run			
Test Risks	This test itself causes no risk to the operation of the application or the			
	existing data in the database.			
Pass/Fail Criteria	Passed: Tableau is successfully connected to MySQL local DB 'gmfsp_db' and			
	shows 'Close' values on the left vertic	•		
	Failed: Tableau is not able to conne	ct to the database a	nd show 'Close'	
	values on the left y-axis.			
Fail Reasons	1 MySQL database is not setup រ	• •		
	2 Tableau data source is not set			
		3 Tableau has encountered an issue(we are limited in correcting		
	Tableau related technical issues as it's a third-party software)			
	4 'FEATURES_IN_TABLEAU_FRL.sql' script to build this tab has issues			
	5 'DataFetch.py' Python script h			
	6 YAHOO data exchange had issues 7 'instrumentstatistics' table has issues in MySQL 'gmfsp db' database			
Driority		Priority Reason		
Priority	High Priority Reason Close price is vital to all visuals		Close price is vital	
Driority Convence	ш 7		to all visuals	
Priority Sequence	H-7			

ID	NFTC_10.0			
Test Case Name	Tableau - Measure names legends mus	st show on each gra	aph	
Created By	Abdul Ahmad	Abdul Ahmad Date Created 11-24-2019		
Description	Tableau - Measure names legends mus	st show on each gra	aph	
Tester	Data Analyst or Developer			
<b>Test Frequency</b>	Anytime the application testing is requ	ıired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly an	d required drivers a	are installed	
Postconditions	1 Tableau will be connected to the d	atabase		
	2 Tableau will be able to pull all data	and show the upd	ated tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file	name	Measure Names	
	'GM_FinTech_Application'		Close	
	2 Double click to open it		highfrllinelong lowfrllinelong	
	3 Click on the <b>'FRLFeatures'</b> tab		Ipeak	
	4 On the right side of the screen, look for a box labeled ltrough			
	'Measure Names'			
Expected Result	Measure Names, which are legend codes show next to each graph on the			
/	right-hand side			
Test Entry/Exit	Entry: Test can be executed at any time in any order on any tab			
Criteria	Exit: When the test completes, or it can be terminated during the run			
Test Risks	This test itself causes no risk to the operation of the application or the			
Barrier 1 Carrier	existing data in the database.			
Pass/Fail Criteria	Passed: Tableau is successfully connected to MySQL local DB 'gmfsp_db'			
	and line graphs for different values are built accurately  Failed: Tableau is not able to connect to the database and show graph			
	measure legend names	i to the database a	nu snow grapn	
Fail Reasons	MySQL database is not setup prope	rly		
raii Neasulis	2 Tableau data source is not setup prope	•	ho data source	
	• •	• •		
		3 Tableau has encountered an issue(we are limited in correcting Tableau related technical issues as it's a third-party software)		
	4 <b>'FEATURES_IN_TABLEAU_FRL.sql'</b> script to build this tab has issues			
	5 Python script 'EngineeredFeatures.py' has issues			
Priority	High	Priority Reason	Legends are	
,	ľ	,	important for all	
			the graphs	
<b>Priority Sequence</b>	H-8			
<b>Priority Sequence</b>	H-8			

ID	NFTC_11.0		
Test Case Name	Tableau - All portofolio values graphs must start with \$10000		
Created By	Abdul Ahmad Date Created 11-24-2019		
Description	Tableau - All portofolio values graphs r	must start with \$10	000
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is requ	iired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	sword = password
	2 Tableau must be setup properly		
	3 Database must be setup properly and	•	are installed
Postconditions	1 Tableau will be connected to the d		
	2 Tableau will be able to pull all data	and show the upd	ated tabs
	3 Application will be functional		
Test Steps	1 Browse to Tableau application file		Custom Strat Runn [FRL,CMA,EMA,MACD
	name 'GM_FinTech_Application'	15K	[FRICHIA, HIA, WARE
	2 Double click to open it	10K- CARZ 5K-	
	3 Click on the tab 'PortfolioCOMB'	OK 15K-	~
	4 Hover the mouse pointer to the far		
	left of any of the <b>'Green'</b> lines	OK instrumentnar portfoliovalue	me: GM
5 . I.D. II	5 A tooltip with 3 values will appear		
Expected Result	The tooltip must show a value of <b>10,000</b> in 'portfoliovalue' data item		
Test Entry/Exit	<b>Entry :</b> Test can be executed at any time in any order on any of the Portfolios		
Criteria Toot Bioles	Exit: When the test completes, or it can be terminated during the run  This test itself causes no risk to the operation of the application or the		
Test Risks	This test itself causes no risk to the operation of the application or the		
Pass/Fail Criteria	existing data in the database.  Passed: Tableau is successfully connected to MySQL local DB 'gmfsp_db'		
Pass/Fail Citteria	and line graphs for 'portfoliovalue'	ted to MysQL loca	I DB gillisp_ub
	<b>Failed</b> : Tableau is not able to connec	t to the database a	nd null correct
	'portfoliovalue' from 3 years ago	t to the database a	na pan correct
Fail Reasons	1 MySQL database is not setup prope	rlv	
Tun Reasons	2 Tableau data source is not setup proper	•	ne data source
	3 Tableau has encountered an issue(v	• • • • • • • • • • • • • • • • • • • •	
	related technical issues as it's a third-party software)		
	4 'PORTFOLIO_VALUE_IN_TABLEAU_		ias issues
	5 <b>'TradingSimulator.py'</b> Python script has issues		
	6 'DateDim' table in the 'gmfsp_db' i	MySQL instance has	s issues
Priority	High	<b>Priority Reason</b>	Portfolio's initial
			value is \$10,000
<b>Priority Sequence</b>	H-9		

ID	NFTC_12.0			
Test Case Name	Tableau - CMA-FRL-Signals tab has multiple instrument name radio toggle			
	buttons functional			
Created By	Abdul Ahmad Date Created 11-24-2019			
Description	Tableau - CMA-FRL-Signals tab has multiple instrument name radio toggle			
	buttons functional. This is a dashboard	tab not a single da	ata tab.	
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>User</b>	name = root Pass	word = password	
	2 Tableau must be setup properly and o	database is configu	red correctly	
Postconditions	1 Tableau will be connected to the da	tabase		
	2 Tableau will be able to pull all data	and show the upda	ited tabs	
	3 Application will be functional			
Test Steps	1 Browse to Tableau application file n	ame	Features instrumentna	
	'GM_FinTech_Application'		- 42 Null CARZ GM	
	2 Double click to open it		28 % XPH	
		B Click on <b>'CMA-FRL-Signals'</b> tab		
	4 Look at the right or left toggle button panels			
	5 Separate toggle radio button stacks should appear(4 in total)			
	6 Click on different options to change graphs			
Expected Result	Graph values must change if the tester changes radio button selections			
Test Entry/Exit	<b>Entry</b> : Test can be executed at any tim	•		
Criteria	<b>Exit:</b> When the test completes, or it can be terminated during the run			
Test Risks	This test itself causes no risk to the operation of the application or the			
	existing data in the database.			
Pass/Fail Criteria	Passed: Tableau is successfully connec	•	I DB <b>'gmfsp_db'</b>	
	and Tableau dashboard page is built co	•		
- "-	Failed: Tableau not able to connect t		radio buttons	
Fail Reasons	1 MySQL database is not setup propo	•	tha data as as	
	2 Tableau data source is not setup pi			
	3 Tableau has encountered an issue(we are limited in correcting Tableau)			
	4 Issues with multiple SQL scripts the		• •	
	5 Issues with multiple Python scripts that are used for different			
Driority	calculations combined on this page			
Priority	Medium	Priority Reason	Radio buttons for	
Driority Coguence	M-3		each graph	
Priority Sequence	IVI-3			

ID	NFTC_13.0		
Test Case Name	Tableau - EMA-MACD-CUSTOM-Signals tab has multiple instrument name		
	radio toggle buttons functional		
Created By	Mohamad Saab Date Created 11-24-2019		
Description	Tableau - EMA-MACD-CUSTOM-Signals tab has multiple instrument name		
	radio toggle buttons functional. This tab is a dashboard, not a single data tab.		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is required		
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Username</b> = root <b>Password</b> = password		
	2 Tableau must be setup properly		
	3 Database must be setup properly and required drivers are installed		
Postconditions	1 Tableau will be connected to the database		
	Tableau will be able to pull all data and show the updated tabs		
	3 Application will be functional		
Test Steps	1 Browse to Tableau application file name		
	'GM_FinTech_Application'		
	2 Double click to open it		
	3 Click on 'EMA-MACD-CUSTOM-Signals' tab		
	4 Look at the right side for toggle button panels		
	5 Separate toggle radio button stacks should appear(3 in total)		
	6 Click on different options to change graphs		
Expected Result	Graph values must change if the tester changes radio button selections		
Test Entry/Exit	Entry: Test can be executed at any time in any order on this tab		
Criteria	<b>Exit:</b> When the test completes, or it can be terminated during the run		
Test Risks	This test itself causes no risk to the operation of the application or the		
- /- U.S.	existing data in the database.		
Pass/Fail Criteria	Passed: Tableau is successfully connected to MySQL local DB 'gmfsp_db'		
	and the Tableau tab is configured accurately.		
Fail Danson	Failed: Tableau is not able to connect to the database and show graph		
Fail Reasons	1 MySQL database is not setup properly		
	2 Tableau data source is not setup properly, recreated the data source		
	3 Tableau has encountered an issue(we are limited in correcting Tableau)		
	4 Multiple SQL scripts that built this view have issues		
	5 'BuySell.py' Python script has issues 6 'DataForecast.py' function 'calculate arima forecast()' has issues		
Priority	Medium Priority Reason Dynamic graphs		
•	M-4		
Priority Sequence	101-4		

ID	NFTC_14.0		
Test Case Name	Tableau - Portfolio-CMA_FRL-EMA tab show data grid with values in each		
	array, intersection		
Created By	John Gettel Date Created 11-24-2019		
Description	Tableau - Portfolio-CMA_FRL-EMA tab show data grid with values in each		
	array, intersection		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is required		
Executed By	Application User		
Preconditions	1 Database is created with MySQL <b>Username</b> = root <b>Password</b> = password		
	2 Tableau must be setup properly		
	3 Database must be setup properly with required drivers installed		
Postconditions	4 The user will be able to view portfolio returns for each strategy and each		
=	symbol		
Test Steps	1 Browse to Tableau application  Strategy  Mago RoyHold CMA COMB EMA FRL MACO		
	file name		
	'GM_FinTech_Application' 1,385 -842 -205 -961 -471 -28 -1,686		
	2 Double click to open it -821 -922 -1,328 -804 -621 -62 572 3 Click on the tab named -77 1,299 777 620 1,043 912 184 645		
	'Portfolio-CMA_FLR-EMA'		
Expected Result	A grid is displayed that gives the total returns for each strategy and each		
Expected Result	ticker symbol		
Test Entry/Exit	Entry: Test can be executed at any time in any order		
Criteria	<b>Exit:</b> When the test completes, or it can be terminated during the run		
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database.		
Pass/Fail Criteria	Passed: Tableau is successfully connected to MySQL local database		
	'gmfsp_db' and data on tab changes appropriately		
	Failed: Tableau is not able to connect to the database, incomplete data, or		
	no data at all is displayed		
Fail Reasons	1 MySQL database is not setup properly		
	2 Tableau data source is not setup properly, recreated the data source		
	3 Tableau has encountered an issue(we are limited in correcting		
	Tableau related technical issues as it's a third-party software)		
	4 SQL Script to build this tab has issue		
Priority	High Priority Reason Viewing portfolio		
	returns is required		
Priority Sequence	H-10		

ID	NFTC_15.0			
Test Case Name	Tableau – Portfolio-CMA_FRL-EMA tab has instrument name radio toggle			
	buttons functional			
Created By	John Gettel Date Created 11-24-2019			
Description	Tableau – Portfolio-CMA_FRL-EMA tab has ins	strumer	nt name radio toggle	
	buttons functional			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is required			
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Username</b>	= root	Password = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly with requi	ired driv	vers installed	
Postconditions	1 User will be able to switch view betwe	een each	n instrument tracked by	
	the application			
Test Steps	1 Browse to Tableau application file	Current Po	rtfolio Value × strateyycoue	
	name 'GM_FinTech_Application'		(All)     algo     BuyHold	
	2 Double click to open it	71,979	71,693	
		3 Click on the tab named 'Portfolio-		
	CMA_FRL-EMA'	94,728	71,875	
	_	4 On the right, there should be list of		
	_	radio buttons		
Formanda di Dancolt	5 Make a different selection other than the		•	
Expected Result	Toggle buttons are filtering data as per design	n. The gr	raph should change	
Test Entry/Exit	when the tester toggles the button.  Entry: Test can be executed at any time in any	v ordor		
Criteria	<b>Exit:</b> When the test completes, or it can be te	•	od during the run	
Test Risks	This test itself causes no risk to the operation			
Test Nisks	existing data in the database.	OI LITE &	application of the	
Pass/Fail Criteria	Passed: Tableau is successfully connected to I	MySOL I	local database	
r ussyrum criteria	'gmfsp db' and data on tab changes appropri	•	ocai database	
	<b>Failed:</b> Tableau is not able to connect to the d	•	or no changes occur	
	on tab		o or the errainges coom.	
Fail Reasons	MySQL database is not setup properly			
	2 Tableau data source is not setup properly,	. recreat	ted the data source	
	3 Tableau has encountered an issue (we are			
	4 SQL Script to build this tab has issue			
	5 Filters setting in Tableaus is not setup properly			
Priority	High Priority Reason Viewing data for			
			symbols is required	
<b>Priority Sequence</b>	H-11			

ID	NFTC_16.0		
Test Case Name	Tableau - Portfolio-MACD-COMB-BuyHold tab show data grid with values in		
	each array, intersection		
Created By	John Gettel	<b>Date Created</b>	11-24-2019
Description	Tableau - Portfolio-MACD-COMB-	BuyHold tab show o	data grid with values in
	each array, intersection		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is	required	
Executed By	Application User		
Preconditions	1 Database is created with MySQL		Password = password
	2 Tableau must be setup properly		
	3 Database must be setup proper	•	
Postconditions	The user will be able to view porti	folio returns for eac	ch strategy and each
	symbol	<u> </u>	
Test Steps	1 Browse to Tableau application	algo BuyHold	Strategy CMA COMS EMA FRL MACD
	name 'GM_FinTech_Application		590 105 486 428 -768
	2 Double click to open it	GM 1,385 -842	-205 -961 -471 -28 -1,686 -1,328 -804 -621 -62 572
	3 Click on the tab named 'Portfo	SPY 1,299 777	620 1,043 912 184 645
	MACD-COMB-BuyHold'	XFH 152 -43	249 15 260 909 189
Expected Result	A grid is displayed that gives the total returns for each strategy and each		
	ticker symbol		
Test Entry/Exit	Entry: Test can be executed at an	y time in any order	
Criteria	Exit: When the test completes, or	it can be terminate	ed during the run
Test Risks	This test itself causes no risk to the operation of the application or the		
	existing data in the database.		
Pass/Fail Criteria	Passed: Tableau is successfully co	nnected to MySQL	local database
	'gmfsp_db' and data on tab chang		
	Failed: Tableau is not able to conr	nect to the database	e, incomplete data, or
	no data at all is displayed		
Fail Reasons	1 MySQL database is not setup p		
	2 Tableau data source is not set		
	3 Tableau has encountered an issue (we are limited in correcting it)		
Dui auth	4 SQL Script to build this tab has		Minutes as afelia
Priority	High	Priority Reason	Viewing portfolio
			returns is a main feature of the
	application		
Priority Sequence	H-12		application
Filolity Sequence	П-14		

ID	NFTC_17.0		
Test Case Name	Tableau – Portfolio-MACD-COMB-BuyHold tab has instrument name radio		
	buttons functional		
Created By	John Gettel	Date Created	11-24-2019
Description	Tableau – Portfolio-MACD-COMB-E	BuyHold tab has in:	strument name radio
	buttons functional		
Tester	Data Analyst or Developer		
Test Frequency	Anytime the application testing is r	equired	
Executed By	Application User		
Preconditions	1 Database is created with MySQL	<b>Username</b> = root	<b>Password</b> = password
	2 Tableau must be setup properly		
	3 Database must be setup properly		
Postconditions	5 User will be able to switch view	between each ins	strument tracked by
	the application		
Test Steps	1 Browse to Tableau application f	ile name <b>'GM_Fin</b> '	Tech_Application'
	2 Double click to open it	چ cent Po	Show Me
	3 Click on the tab named 'Portfol	io-MACD-	(All) algo BuyHold
	COMB-BuyHold'		71,693
	4 On the right, there should be lis	t of radio	● FRL ● MACD
	<ul><li>buttons</li><li>Make a different selection other than the default selected option</li></ul>		
			<u> </u>
Expected Result	Toggle buttons are filtering data as per design. The graph should change		
Table 15 /F 11	when the tester toggles the button		
Test Entry/Exit	Entry: Test can be executed at any	•	مريس مطلع مينسان المالم
Criteria	<b>Exit:</b> When the test completes, or i		=
Test Risks	This test itself causes no risk to the	operation of the a	application or the
Doce/Foil Critoria	existing data in the database.	nostad to Muscal	lacal databasa
Pass/Fail Criteria	Passed: Tableau is successfully con 'gmfsp_db' and data on tab change	•	ocai database
	Failed: Tableau is not able to conne		or no changes occur
	on tab	ect to the database	e of the changes occur
Fail Reasons	MySQL database is not setup pr	onerly	
I all iteasons	2 Tableau data source is not setup pr		ted the data source
	3 Tableau has encountered an iss		
	4 SQL Script to build this tab has i	•	• ,
	5 Filters setting in Tableaus is not setup properly		
Priority	High	Priority Reason	Viewing data for
			different symbols is
			required
<b>Priority Sequence</b>	H-13		

ID	NFTC_18.0			
Test Case Name	Tableau – Small play and reverse buttons on the bottom right of the screen			
	are functional and move the tabs to th	e right or left		
Created By	John Gettel	John Gettel Date Created 11-24-2019		
Description	Tableau – Small play and reverse butto	ons on the bottom	right of the screen	
	are functional and move the tabs to th	e right or left		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	iired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	<b>sword</b> = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly wit	·		
Postconditions	1 User will be able to scroll through the	he menu of sheets	at the bottom of	
	the tableau application			
Test Steps	1 Browse to Tableau application file n	name <b>'GM_FinTech</b>	_Application'	
	2 Double click to open it			
	3 On the lower right	Portfolio	FRL II	
	hand corner of the	14 4		
	screen click the 'play' button to view the next sheet in the list			
	4 Click the 'fast forward' button to scroll to the last sheet			
	5 Click the 'reverse' button to view the previous sheet			
	6 Click the 'rewind' button to scroll back to the first sheet 7 Each of the above steps should allow the user to view a new sheet and			
	7 Each of the above steps should allow the user to view a new sheet and click on it to view			
Expected Posult	Each of the four scroll buttons should	displace the sheets	har by a prodofined	
Expected Result	distance configured by Tableau	displace the sheets	bar by a predefined	
Test Entry/Exit	Entry: Test can be executed at any tim	e in any order		
Criteria	<b>Exit:</b> When the test completes, or it ca		ıring the run	
Test Risks	-			
rese misits	This test itself causes no risk to the operation of the application or the existing data in the database.			
Pass/Fail Criteria	Passed: A new sheet is available to be	clicked on by the u	ser	
	Failed: No new sheets appear unless a			
	sheets	0 0		
Fail Reasons	1 MySQL database is not setup p	roperly		
	2 Tableau data source is not setu	ip properly, recreat	ed the data source	
	3 Tableau has encountered an iss	sue (we are limited	in correcting it)	
	4 SQL Script to build each individual tab has failed to execute properly			
Priority	Medium Priority Reason Viewing all data			
			tabs is optimal	
<b>Priority Sequence</b>	M-5			

ID	NFTC_19.0			
Test Case Name	Tableau - CMAFeatures tab shows close, Icma, scma and wcma values and			
	radio buttons for the instrument name	9		
Created By	John Gettel Date Created 11-24-2019			
Description	Tableau - CMAFeatures tab shows clos	se, Icma, scma and	wcma values and	
	radio buttons for the instrument name	2		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ıired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	<b>sword</b> = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly wi	th required drivers	installed	
Postconditions	User will be able to view the close, Icm	na, scma, and wcma	a values on each	
	date for each individual symbol			
Test Steps	1 Browse to Tableau application file	name 'GM_FinTec	h_Application'	
	2 Double click to open it	Cross	Moving Average Features natumentonic	
	3 On the right side of the application		40 • (41 • (42	
	each radio button for the ticker sy		33 0 CM 0 FF 0 FF 0 977 0 397	
	The data showing on the the graph should			
	change for each symbol			
	2 Each symbol should have a lema, sema, wella,			
	and close data point for each day.			
Expected Result	The data displayed in each graph shou	id change for each	ticker symbol	
T F /F !!	selected with a radio button			
Test Entry/Exit	Entry: Test can be executed at any tim		t. a. th. a	
Criteria	<b>Exit:</b> When the test completes, or it can be terminated during the run			
Test Risks	This test itself causes no risk to the op	eration of the appli	ication or the	
Doce/Fail Critoria	existing data in the database.	a appropriate data	on each day for	
Pass/Fail Criteria	Passed: The user can select and see th	e appropriate data	on each day for	
	each symbol  Failed: No data or incomplete data applements.	noars or no togglin	og occure	
Fail Reasons	1 MySQL database is not setup prop		ig occurs	
raii Neasoiis	1 ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' '	•		
	2 Tableau data source is not setup properly			
	<ul> <li>Tableau has encountered an issue (we are limited in correcting Tableau)</li> <li>SQL Script to build each individual tab has failed to execute properly</li> </ul>			
Priority	Medium Priority Reason Technical data			
THOTICY	Medium	Thority Reason	view is a good	
			feature	
Priority Sequence	M-6		- Catalo	
. Hority sequence				

ID	NFTC_20.0			
Test Case Name	Tableau - FRLFeatures tab shows Close, five different FRL values and radio			
	buttons for the instrument name			
Created By	John Gettel	Date Created	11-24-2019	
Description	Tableau - FRLFeatures tab shows Close	, five different FRL	values and radio	
	buttons for the instrument name			
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	rname = root Pas	<b>sword</b> = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly wit	th required drivers	installed	
Postconditions	Users will be able to view the close pri	ce and each of the	five different	
	Fibonacci Retracement levels for each		•	
Test Steps	1 Browse to Tableau application file n	name <b>'GM_FinTech</b>	_Application'	
	2 Double click to open it	Fibonacci Retracement Features	instrumentname	
	3 On the right side of the application		CARZ  GM  PTE	
	click through each radio button for		◆ SPY ◆ XPH	
	l •	the ticker symbols		
	4 The data showing on the the graph			
	should change for each symbol			
	5 Each symbol should have five different Fibonnacci retracement levels for each date			
Formanda d Danielt	The data displayed in each graph should change for each ticker symbol			
Expected Result	selected with a radio button	id change for each	ticker symbol	
Tost Entry/Evit		o in any order		
Test Entry/Exit Criteria	Entry: Test can be executed at any time in any order  Exit: When the test completes, or it can be terminated during the run			
Test Risks	·			
Test Nisks	This test itself causes no risk to the operation of the database.	eration of the appli	ication of the	
Pass/Fail Criteria	Passed: The user can click through and	I see the annronria	te data on each day	
1 assyran Criteria	for each symbol	i see the appropria	te data on each day	
	<b>Failed:</b> No data or incomplete data app	nears or no toggling	g occurs	
Fail Reasons	1 MySQL database is not setup prope		<u> </u>	
	2 Tableau data source is not setup pro	•		
	3 Tableau has encountered an issue (we are limited in correcting it)			
	4 SQL Script to build each individual tab has failed to execute properly			
Priority	Medium	<b>Priority Reason</b>	Viewing technical	
			data is a good	
			feature	
<b>Priority Sequence</b>	M-7			

ID	NFTC_21.0			
Test Case Name	Tableau - Tableau - BollingerBands tab	shows Close, lowe	r and, upper bound	
	values appropriately and radio buttons	s for the instrumen	t name	
Created By	Aalem Singh	Date Created	11-24-2019	
Description	Tableau - BollingerBands tab shows Cl			
	appropriately and radio buttons for th	e instrument name		
Tester	Data Analyst or Developer			
Test Frequency	Anytime the application testing is requ	ired		
Executed By	Application User			
Preconditions	1 Database is created with MySQL <b>Use</b>	ername = root Pas	sword = password	
	2 Tableau must be setup properly			
	3 Database must be setup properly wi	·		
Postconditions	User will be able to view the close pric			
Table Class	Fibonacci Retracement levels for each		•	
Test Steps	1 Browse to Tableau application file	name 'GIVI_FINTeci	n_Application'	
	2 Double click to open it	Bollinger Bands	InstrumentName  ■ (All)	
	3 On the right side of the application		● Null ● CARZ	
	click through each radio button for			
	the ticker symbols  4. The data showing on the the graph			
	4 The data showing on the the graph should change for each symbol			
	5 Each symbol should have an upper			
	and a lower Bolligner Band			
	displayed for each date			
Expected Result	The data displayed in each graph should change for each ticker symbol			
/	selected with a radio button			
Test Entry/Exit	Entry: Test can be executed at any tim			
Criteria	<b>Exit:</b> When the test completes, or it ca		-	
Test Risks	This test itself causes no risk to the op	eration of the appli	cation or the	
Dece/Feil Cuiterie	existing data in the database.	J dot	dov. for on the summer of	
Pass/Fail Criteria	Passed: The user can click through and			
Foil Dessens	Failed: No data or incomplete data app		goccurs	
Fail Reasons	<ol> <li>MySQL database is not setup p</li> <li>Tableau data source is not setu</li> </ol>			
	<ul><li>2 Tableau data source is not setu</li><li>3 Tableau has encountered an is:</li></ul>		in correcting it)	
	4 SQL Script to build each individ	•	<b>O</b> ,	
Priority	Medium	Priority Reason	Viewing technical	
THOTILY	Wediani	Thority Reason	data is important	
Priority Sequence	M-8		aata is iiiportant	
1 Hority Sequence	141.0			

### **4 PRODUCT TESTING CERTIFICATION**

The undersigned acknowledge that they have overseen or themselves performed each test listed in this document for the GM FinTech Tableau/PyCharm/MySQL based application and all tests were successful and passed the expected results criteria.

Team Lead, Front End Lead, Back End Lead, QA Lead(if available) Signature Date **Print Name** Title Role **Signature** Date **Print Name Title** Role **Signature Date Print Name** Title Role **Signature Date Print Name Title** Role

#### **APPENDIX**

### **APPENDIX A: SQL AND PYTHON TESTING SCRIPTS**

```
Test Case Name

Database Engine has the correct MYSQL connection string

Created By

Abdul Ahmad

Date Created

# TC_1.0_Database Engine

import sqlalchemy as sal
from FinsterTab.F2019.dbEngine import DBEngine

conn_str = 'mysql+pymysql://root:password@localhost:3306/gmfsp_db'
actual_engine = DBEngine().mysql_engine()
expected_engine = sal.create_engine(conn_str)

if actual_engine == expected_engine:
    print("correct engine")

else:
    print("incorrect engine")
```

ID	TC_2.0		
Test Case Name	Correct financial data exchange for fetching base statistics		
<b>Created By</b>	Abdul Ahmad	<b>Date Created</b>	11-10-2019
from FinsterTab.F2 conn_str = 'mysql+  engine = DBEngine( db_engine = DBEng instrument_master master_data = Data if master_data.data	019.dbEngine import DBEngine 019.DataFetch import DataFetch pymysql://root:password@localhost:330( ().mysql_engine() gine().mysql_engine() = 'dbo_instrumentmaster' aFetch(engine, instrument_master)		

ID	TC_3.0				
<b>Test Case Name</b>	No duplicate instrument ids in 'instrumentmaster' table				
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019		
Script ID : TC_3	3.0				
SELECT instrumentid, count(instrumentname) cnt					
FROM dbo_instrum	FROM dbo_instrumentmaster as m				
group by instr having count(instru	rumentid Imentname) > 1				

ID	TC_4.0				
<b>Test Case Name</b>	No duplicate values/records in future forecast for each symbol				
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019		
Script ID : TC_4	4.0				
SELECT a.instrumentid, count(distinct a.					
FROM dbo_algorith	nmforecast as a				
where a.forecastdate > current_date()					
group by a.instrumentid;					

ID	TC_5.0			
Test Case Name	No duplicate values/records in past forecast for each symbol			
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019	
Script ID : TC_	5.0			
SELECT a.instrumentid, count(distinct a.forecastdate) cnt				
FROM dbo algorith	nmforecast as a			
where a.forecastdate < current_date()				
group by				
a.instrumentid				
;				

ID	TC_6.0				
Test Case Name	No duplicate values/records in past forecast for each symbol				
Created By	Mohamad Saab	<b>Date Created</b>	11-22-2019		
Script ID : TC_6.0					
use GMFSP_db;	use GMFSP_db;				
SELECT instrument	id InstrumentID,				
	forecastcloseprice) ForecastPriceAverage	,			
min(forecastda	•				
max(forecastd	•				
•	forecastdate) NumOfDaysAhead				
	FROM dbo_algorithmforecast WHERE forecastdate > current date()				
GROUP BY instrumentid					
GROOT BY INSTIGUTE	Citia				

ID	TC_7.0		
<b>Test Case Name</b>	No duplicate values/records in past forecast for each symbol		
Created By	Mohamad Saab	<b>Date Created</b>	11-21-2019
Script ID : TC_7.	0		
use GMFSP_db;			
SELECT algorithmc	ode AlgorithmName,		
avg(	forecastcloseprice) ForecastPriceAverage	,	
min(forecastda	•		
max(forecastd	•		
	forecastdate) Num Of Days Ahead		
FROM dbo_algorithmforecast			
WHERE forecastdate > current_date()			
GROUP BY algorith	mcode		

ID	TC_8.0			
<b>Test Case Name</b>	No duplicate values/records in past forecast for each symbol			
Created By	Abdul Ahmad Date Created 11-21-2019			
Script ID : TC_	8.0			
SELECT count(distinct instrumentid) instid, count(distinct forecastdate) date, avg( forecastcloseprice) avgprice				
FROM dbo_algorith				
left outer join (	left outer join dbo_datedim as d on a.forecastdate = d.date			
where forecastdate < current_date() and d.isholiday = 0 and d.weekend = 0				

ID	TC_9.0			
<b>Test Case Name</b>	Past forecast values must exist for last 3	years until today for	or each algorithm	
Created By	Abdul Ahmad	Date Created	11-21-2019	
Script ID : TC_9	9.0			
SELECT count(distinct algorithmcode) algo, count(distinct forecastdate) date, avg(forecastcloseprice) avgprice				
FROM dbo_algorithmforecast as a left outer join dbo_datedim as d on a.forecastdate = d.date				
where forecastdate < current_date() and d.isholiday = 0 and d.weekend = 0 ;				

```
ID
                   TC 10.0
Test Case Name
                   Ten future days from today with no prediction error for each symbol
                                                          Date Created
                   Abdul Ahmad
                                                                            11-21-2019
Created By
             : TC 10.0 --
-- Script ID
SELECT
   count(distinct forecastdate) date,
   avg(prederror)
                         prederror
FROM dbo_algorithmforecast as a
    left outer join dbo_datedim as d on a.forecastdate = d.date
where forecastdate > current_date()
 and d.isholiday = 0
 and d.weekend = 0
```

ID	TC_11.0			
<b>Test Case Name</b>	Ten future days from today with no pred	liction error for eac	ch algorithm	
<b>Created By</b>	Abdul Ahmad	Abdul Ahmad Date Created 11-22-2019		
Script ID : TC_3	11.0			
SELECT count(distinct forecastdate) date, count(distinct algorithmcode) algo, avg(prederror) prederror				
FROM dbo_algorithmforecast as a left outer join dbo_datedim as d on a.forecastdate = d.date				
<pre>where forecastdate &gt; current_date() and d.isholiday = 0 and d.weekend = 0 ;</pre>				

```
Test Case Name
Nine tables exist in the database

Created By
Abdul Ahmad
Date Created

-- Script ID: TC_12.0 --

use gmfsp_db;

SELECT COUNT(*) as cnt
FROM information_schema.tables
WHERE table_schema = 'gmfsp_db' and table_name like 'dbo_%'
;
```

ID	TC_13.0		
<b>Test Case Name</b>	ARIMA has first 9 days blank		
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019
Script ID : TC_13.	0		
use gmfsp_db;			
SELECT i.date,			
i.instrumentid,			
i.close,			
a.algorith	mcode,		
a.forecastclose	price		
FROM dbo_instrum	nentstatistics as i		
left outer join	dbo_algorithmforecast as a on i.instrume	entid = a.instrumen	tid and
	<pre>i.`date` = a.forecastdate and</pre>		
	a.algorithmcode ='arima'		
order by i.date			
;			

ID	TC_14.0			
Test Case Name	Random Forest has first 9 days blank			
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019	
Script ID : TC_14.	0			
use gmfsp_db;				
SELECT i.date,				
i.instrumentid,				
i.close,				
a.algorith	mcode,			
a.forecastclose	price			
FROM dbo_instrum	nentstatistics as i			
left outer join dbo_algorithmforecast as a on i.instrumentid = a.instrumentid and i.`date` = a.forecastdate and a.algorithmcode ='randomforest'				
order by i.date ;				

ID	TC_15.0			
Test Case Name	No missing values in the 'close' or 'date' column in the instrumentstatistics			
	table			
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019	
Script ID : TC_15.	0			
use gmfsp_db;	e gmfsp_db;			
SELECT i.`date`,				
i.instrumentid,				
i.`close`				
FROM dbo_instrumentstatistics as i				
where i.`close` is null or i.date is null				
;				

ID	TC_16.0			
Test Case Name	BuySell signals must exist for all reccords for past 3 years, values of -1,0,1 for			
	each day			
Created By	Abdul Ahmad	<b>Date Created</b>	11-22-2019	
Script ID : TC_16.	0			
use gmfsp_db;  SELECT i.`date`,  count(distinct i.signal) signals				
_	FROM dbo_actionsignals as i			
group by i.date	- · ·			
order by i.date				
;				

ID	TC_17.0			
<b>Test Case Name</b>	BuySell signals must exist for each strategy			
Created By	Abdul Ahmad	Date Created	11-22-2019	
Script ID : TC_17.	0			
use gmfsp_db;	use gmfsp_db;			
SELECT i.`date`,				
count(distinct i	.strategycode) strcd,			
count(distinct i	count(distinct i.signal) signals			
FDOM the continue to the continue to				
FROM dbo_actionsignals as i				
	group by i.date			
order by i.date				
,				

ID	TC_18.0			
<b>Test Case Name</b>	BuySell signals must exist for each symbol			
<b>Created By</b>	Abdul Ahmad	<b>Date Created</b>	11-22-2019	
Script ID : TC_18.	0			
use gmfsp_db;	use gmfsp_db;			
· ·	SELECT i.`date`, count(distinct i.instrumentid) instid, count(distinct i.signal) signals			
FROM dbo_actionsignals as i group by i.`date` having count(distinct i.instrumentid) <> 5 ;				

ID	TC_19.0			
<b>Test Case Name</b>	Six algorithm codes exist in algorithmmaster table			
<b>Created By</b>	Abdul Ahmad	Abdul Ahmad Date Created 11-22-2019		
Script ID : TC_19.	0			
use gmfsp_db;				
SELECT *				
FROM dbo_algorithmmaster				
;				

```
Test Case Name Instrumentmaster must have only 5 stock symbols

Created By Abdul Ahmad Date Created 11-22-2019

-- Script ID : TC_20.0 --

use gmfsp_db;

SELECT *

FROM dbo_instrumentmaster
;
```

ID	TC_21.0			
Test Case Name	Instrumentstats column named 'high' has values >= 'low' , 'open', 'close'			
	columns on all records			
Created By	Abdul Ahmad	Date Created	11-23-2019	
Script ID : TC_21.	0			
use gmfsp_db;	use gmfsp_db;			
SELECT *				
FROM dbo_instrumentstatistics				
where				
_	high < low or			
high < `open` or	•			
high < `close`				
<b> </b> ;				

```
TC_22.0
                  Instrumentstats column names 'low' has values <= high, open, close columns
Test Case Name
                  on all records
                                                         Date Created
Created By
                  Abdul Ahmad
                                                                           11-23-2019
-- Script ID : TC_22.0 --
use gmfsp_db;
SELECT *
FROM dbo_instrumentstatistics
where
       low > high or
  low > `open` or
  low > `close`
```

ID	TC_23.0		
<b>Test Case Name</b>	Instrumentstats column 'volume' values	should always be >	>=0
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019
Script ID : TC_23.	0		
use gmfsp_db;			
SELECT *			
FROM dbo_instrumentstatistics			
where volume < 0;			

ID	TC_24.0		
<b>Test Case Name</b>	Instrumentstats must have 3 years of data till today, going back 3 years from		
	today		
Created By	Abdul Ahmad	Date Created	11-23-2019
Script ID : TC_24.	0		
use gmfsp_db;	gmfsp_db;		
SELECT min(date) startdate, max(date) enddate			
FROM dbo_instrumentstatistics			
;			

ID	TC_25.0			
<b>Test Case Name</b>	Datedim must contain 50 future days from today			
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019	
Script ID : TC_25.	0			
use gmfsp_db;	use gmfsp_db;			
SELECT min(date) todaysdate, max(date) tendaysintofuture, datediff(cast(date_add(now(), interval 50 day) as date), current_date()) daysdiff				
FROM dbo_datedim				
<pre>WHERE      `date` &gt;= current_date() and      `date` &lt; cast(date_add(now(), interval 50 day) as date) ;</pre>				

```
ID
                   TC 26.0
                   Datedim must contain last 3 years of dates, up till today
Test Case Name
Created By
                   Abdul Ahmad
                                                           Date Created
                                                                             11-23-2019
-- Script ID : TC 26.0 --
use gmfsp_db;
SELECT min(date) todaysdate,
    max(date) tendaysintofuture,
    datediff(cast(date_add(now(), interval 3 year) as date), current_date()) daysdiff
FROM dbo_datedim
WHERE
       `date` <= current date() and
  `date` >= cast(date_add(now(), interval -3 year) as date)
```

ID	TC_27.0			
<b>Test Case Name</b>	No duplicate dates or tuples in datedim table			
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019	
Script ID : TC_27.	0			
use gmfsp_db;				
SELECT count(*) cr	nt,			
date	date			
FROM dbo_datedin	FROM dbo_datedim			
GROUP by date				
HAVING count(*) >	HAVING count(*) > 1			
;				

```
TC_28.0

Test Case Name In Statisticalreturns table, cash on hand value must be >=0 for all records

Created By Abdul Ahmad Date Created 11-23-2019

-- Script ID : TC_28.0 --

use gmfsp_db;

SELECT date, instrumentid, strategycode, cashonhand

FROM dbo_statisticalreturns

WHERE cashonhand < 0
;
```

ID	TC_29.0		
Test Case Name	In Statisticalreturns table, 'portfoliovalue' data values must be >= 'cash on		
	hand' values for all records		
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019
Script ID : TC_29.	0		
use gmfsp_db; SELECT `date`, instrumentid, strategycode, cashonhand, portfoliovalue			
FROM dbo_statistic WHERE cashonhand ;			

```
TC_30.0
ID
                 In Statisticalreturns table, 'positionsize' column must be an integer data type
Test Case Name
Created By
                 Abdul Ahmad
                                                      Date Created
                                                                       11-23-2019
-- Script ID : TC 30.0 --
use gmfsp_db;
SELECT TABLE_NAME,
   COLUMN_NAME,
    DATA_TYPE
FROM INFORMATION_SCHEMA.COLUMNS
WHERE table name = 'dbo statisticalreturns' AND
       COLUMN_NAME = 'positionsize'
```

ID	TC_31.0			
<b>Test Case Name</b>	Statisticalreturns table must contain 7 st	rategies		
<b>Created By</b>	Abdul Ahmad	<b>Date Created</b>	11-23-2019	
Script ID : TC_31.	0			
use gmfsp_db;				
SELECT count(distinct strategycode) stratcode				
FROM dbo_statisticalreturns ;				

ID	TC_32.0		
<b>Test Case Name</b>	Fibonacci lines values are in order, 'highfrllinelong' values are >=		
	'lowfrllinelong' values for all records		
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019
Script ID : TC_32.	0		
use gmfsp_db;	use gmfsp_db;		
SELECT *	SELECT *		
FROM dbo_engineeredfeatures			
WHERE lowfrllinelong > highfrllinelong			
;			

ID	TC_33.0			
<b>Test Case Name</b>	Bollinger Bands values are in order, boll	_ub_v > boll_lb_v v	alues for all records	
<b>Created By</b>	Abdul Ahmad	<b>Date Created</b>	11-23-2019	
Script ID : TC_33.	0			
use gmfsp_db;	use gmfsp_db;			
SELECT *				
FROM dbo_engineeredfeatures				
WHERE boll_lb_v >	boll_ub_v			

ID	TC_34.0			
Test Case Name	Each date must have a 0, 1, -1 signal for each strategy and symbol in			
	'actionsignals' table			
Created By	Abdul Ahmad	Date Created	11-23-2019	
Script ID : TC_34.	0			
use gmfsp_db;				
drop table if exists	`a`;			
drop table if exists	`b`;			
drop table if exists	`c`;			
create temporary to	able a			
SELECT `signal`,				
count(`date`) cnt				
FROM dbo_actions	ignals			
GROUP BY 'signal';				
create temporary to				
	Records Grouped by Signal Above' as `sigr	nal`,		
sum(cnt) as cnt				
from a;				
select *				
	from a UNION ALL			
select *				
	from b UNION ALL			
	select 'Total Records in "actionsignals" Table' as `signal`,			
count(*)				
from dbo_actionsig	nals;			

ID	TC_35.0			
<b>Test Case Name</b>	In Statisticalreturns table 'cash on hand'	values must be be	tween 0 and	
	'portfoliovalue', it should never exceed t	he 'portfolio' value		
Created By	Abdul Ahmad	<b>Date Created</b>	11-23-2019	
Script ID : TC_35.	0			
use gmfsp_db;	use gmfsp_db;			
SELECT min(cashonhand) mincoh, max(cashonhand) maxcoh, max(portfoliovalue) maxportfolio				
FROM dbo_statisticalreturns ;				

ID	TC_36.0		
<b>Test Case Name</b>	BuyHold strategy must have no signals, this strategy should only hold the stock		
	and not sell it or buy more		
<b>Created By</b>	Aalem Singh	<b>Date Created</b>	11-23-2019
Script ID : TC_36.	0		
use gmfsp_db;	use gmfsp_db;		
SELECT *	SELECT *		
FROM dbo_actionsignals			
WHERE strategycode = 'buyhold' ;			

ID	TC_37.0		
<b>Test Case Name</b>	In 'actionsignals' table frl, cma, ema, macd signals sum must be between -5		
	and 5		
<b>Created By</b>	Aalem Singh	<b>Date Created</b>	11-23-2019
Script ID : TC_37.	0		
use gmfsp_db;			
select `date`,			
instrumentid,			
sum(`signal`) as	s sgnl		
FROM dbo_actionsignals			
WHERE strategycoo	de in ('frl' , 'cma', 'ema', 'macd')		
group by `date`,	, , , , ,		
instrumentid			
having sum(`signal`) < -5 or			
sum(`signal`) >	5		
;			

ID	TC_38.0		
Test Case Name	getdatasources() function returns correct number of symbols		
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_38.0			
from FinsterTab.F2019.dbEngine import DBEngine from FinsterTab.F2019.DataFetch import DataFetch			
engine = DBEngine().m	engine = DBEngine().mysql_engine()		
fetch = DataFetch(engine, 'dbo_instrumentmaster')			
tickers = fetch.get_datasources()			
<pre>if len(tickers) == 5:     print('incorrect number of symbols: %s' % str(len(tickers)))</pre>			

ID	TC_39.0			
<b>Test Case Name</b>	getdatasources() function returns correct ticker symbols			
Created By	John Gettel	<b>Date Created</b>	11-24-2019	
# TC_39.0				
	from FinsterTab.F2019.dbEngine import DBEngine from FinsterTab.F2019.DataFetch import DataFetch			
fetch = DataFetch(engin	engine = DBEngine().mysql_engine() fetch = DataFetch(engine, 'dbo_instrumentmaster') tickers = fetch.get_datasources()			
symbols = ['GM', 'PFE', 'SPY', 'XPH', 'CARZ']				
,	ols)): s['instrumentname'][i]: r: %s' % symbols[i])			

ID	TC_40.0			
Test Case Name	getdata () function runs successfully			
Created By	John Gettel	<b>Date Created</b>	11-24-2019	
# TC_40.0				
from FinsterTab.F2019.dbEngine import DBEngine from FinsterTab.F2019.DataFetch import DataFetch				
engine = DBEngine().mysql_engine() fetch = DataFetch(engine, 'dbo_instrumentmaster') tickers = fetch.get_datasources() data = fetch.get_data(tickers)				
if data is not None:  print('Incorrect return value while running get_data function: %s' % data)				
,	s['instrumentname'][i]: er: %s' % symbols[i])			

ID	TC_41.0		
<b>Test Case Name</b>	get_calendar() function runs successfully		
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_41.0			
	dbEngine import DBEngine		
from Finster Lab. F2019.	.DataFetch import DataFetch		
engine = DBEngine().mysql_engine()			
fetch = DataFetch(engin	fetch = DataFetch(engine, 'dbo_instrumentmaster')		
calendar = fetch.get_calendar()			
if calendar is not None:			
print('Incorrect return	value while running get_calendar function: %s' %	calendar)	

ID	TC_42.0		
Test Case Name	calculate() function runs successfully		
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_42.0			
from FinsterTab.F2019 engine = DBEngine().m	eatures(engine, 'dbo_instrumentmaster')		
if eng_feat is not None: print('Incorrect return	value while running calculate function: %s' % eng	_feat)	

ID	TC_43.0		
<b>Test Case Name</b>	calculate_forecast() function runs successfully		
<b>Created By</b>	John Gettel	<b>Date Created</b>	11-24-2019
# TC_43.0			
from FinsterTab.F2019.dbEngine import DBEngine from FinsterTab.F2019.DataForecast import DataForecast			
engine = DBEngine().mysql_engine() forecast = DataForecast(engine, 'dbo_instrumentmaster') forecast = forecast.calculate_forecast()			
if forecast is not None: print('Incorrect return	value while running calculate_forecast function: %	s' % forecast)	

ID	TC_44.0		
Test Case Name	calculate_arima_forecast() function runs successfully		
<b>Created By</b>	John Gettel	Date Created	11-24-2019
# TC_44.0			
from FinsterTab.F2019.dbEngine import DBEngine from FinsterTab.F2019.DataForecast import DataForecast			
engine = DBEngine().mysql_engine()			
	forecast = DataForecast(engine, 'dbo_instrumentmaster')		
forecast = forecast.calculate_arima_forecast()			
if forecast is not None:  print('Incorrect return value while running calculate_arima_forecast function: %s' % forecast)			

ID	TC_45.0			
<b>Test Case Name</b>	calculate random forest forecast() function runs successfully			
Created By	John Gettel	John Gettel Date Created 11-24-2019		
# TC_45.0				
from FinsterTab.F2019.dbEngine import DBEngine from FinsterTab.F2019.DataForecast import DataForecast				
engine = DBEngine().mysql_engine()				
forecast = DataForecast(engine, 'dbo_instrumentmaster')				
forecast = forecast.calculate_random_forest_forecast()				
if forecast is not None:  print('Incorrect return value while running calculate_random_forest_forecast function: %s' % forecast)				

ID	TC_46.0		
Test Case Name	calculate_svm_forecast() function runs	successfully	
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_46.0			
	.dbEngine <mark>import</mark> DBEngine .DataForecast <mark>import</mark> DataForecast		
engine = DBEngine().mysql_engine() forecast = DataForecast(engine, 'dbo_instrumentmaster') forecast = forecast.calculate_svm_forecast()			
if forecast is not None: print('Incorrect return	value while running calculate_svm_forecast functi	on: %s' <b>% forecast)</b>	

ID	TC_47.0		
<b>Test Case Name</b>	calculate_forecast_old() function runs si	uccessfully	
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_47.0			
	dbEngine import DBEngine DataForecast import DataForecast		
engine = DBEngine().m forecast = DataForecast forecast = forecast.calc	t(engine, 'dbo_instrumentmaster')		
if forecast is not None: print('Incorrect return	value while running calculate_old_forecast functio	n: %s' % forecast)	

ID	TC_48.0		
<b>Test Case Name</b>	calculate_rxgboost_forecast() function r	uns successfully	
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_48.0			
	dbEngine import DBEngine DataForecast import DataForecast		
engine = DBEngine().mysql_engine()			
forecast = DataForecas	orecast = DataForecast(engine, 'dbo_instrumentmaster')		
forecast = forecast.calculate_xgboost_forecast()			
if forecast is not None:	None:		
print('Incorrect return	value while running calculate_xgboost_forecast fu	nction: %s' % forecast)	

ID	TC_49.0		
<b>Test Case Name</b>	cma_signal() function runs successfully		
<b>Created By</b>	John Gettel	<b>Date Created</b>	11-24-2019
# TC_49.0			
from FinsterTab.F2019. from FinsterTab.F2019.	dbEngine import DBEngine BuySell import BuySell		
engine = DBEngine().m signal = BuySell(engine signal = signal.cma_sig	e, 'dbo_instrumentmaster')		
if signal is not None: print('Incorrect return	value while running cma_signal function: %s' % si	gnal)	

ID	TC_50.0		
Test Case Name	frl_signal() function runs successfully		
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_50.0			
from FinsterTab.F2019.	dbEngine import DBEngine		
from FinsterTab.F2019.	terTab.F2019.BuySell import BuySell		
engine = DBEngine().m	engine = DBEngine().mysql_engine()		
signal = BuySell(engine	e, 'dbo_instrumentmaster')		
signal = signal.frl_signa	I()		
if signal is not None:			
print('Incorrect return	value while running frl_signal function: %s' % sign	al)	

ID	TC_51.0		
<b>Test Case Name</b>	ema_signal() function runs successfully		
Created By	John Gettel	Date Created	11-24-2019
# TC_51.0			
from FinsterTab.F2019. from FinsterTab.F2019.	dbEngine import DBEngine BuySell import BuySell		
engine = DBEngine().mysql_engine()			
signal = BuySell(engine, 'dbo_instrumentmaster')			
signal = signal.ema_sig	ınal()		
if signal is not None: print('Incorrect return	value while running ema_signal function: %s' <b>% s</b> i	gnal)	

ID	TC_52.0		
Test Case Name	macd_signal() function runs successfully	,	
Created By	John Gettel	<b>Date Created</b>	11-24-2019
# TC_52.0			
from FinsterTab.F2019.	dbEngine import DBEngine		
from FinsterTab.F2019.	BuySell import BuySell		
engine = DBEngine().m	ysql_engine()		
signal = BuySell(engine	e, 'dbo_instrumentmaster' <b>)</b>		
signal = signal.macd_si	gnal()		
if signal is not None:			
print('Incorrect return	value while running macd_signal function: %s' % s	signal)	

ID	TC_53.0		
<b>Test Case Name</b>	algo_signal() function runs successfully		
Created By	John Gettel	Date Created	11-24-2019
# TC_53.0			
from FinsterTab.F2019. from FinsterTab.F2019.	dbEngine import DBEngine BuySell import BuySell		
engine = DBEngine().mysql_engine()			
• • •	e, 'dbo_instrumentmaster')		
signal = signal.algo_sig	nai()		
if signal is not None:			
print('Incorrect return	value while running algo_signal function: %s' % si	ignal)	

ID	TC_54.0		
<b>Test Case Name</b>	trade_sim() function runs successfully		
Created By	John Gettel	Date Created	11-24-2019
# TC_54.0			
from FinsterTab.F2019.	dbEngine import DBEngine		
from FinsterTab.F2019.	9.TradingSimulator import TradingSimulator		
engine = DBEngine().mysql_engine()			
simulator = TradingSimulator(engine, 'dbo_instrumentmaster')			
simulator = simulator.tra	ade_sim()		
if simulator is not None:	ulator is not None:		
print('Incorrect return	value while running trade_sim function: %s' % sim	nulator)	

ID	TC_55.0		
<b>Test Case Name</b>	comb_sim() function runs successfully		
Created By	John Gettel	Date Created	11-24-2019
# TC_55.0			
	dbEngine import DBEngine TradingSimulator import TradingSimulator		
engine = DBEngine().m simulator = TradingSim simulator = simulator.co	ulator(engine, 'dbo_instrumentmaster')		
if simulator is not None: print('Incorrect return	: value while running combination_trade_sim function	on: %s' <b>% simulator)</b>	

ID	TC_56.0		
<b>Test Case Name</b>	buy_hold_sim() function runs successful	ly	
Created By	John Gettel	Date Created	11-24-2019
# TC_56.0			
from FinsterTab.F2019.	dbEngine import DBEngine		
from FinsterTab.F2019.	9.TradingSimulator import TradingSimulator		
engine = DBEngine().mysql_engine()			
simulator = TradingSimulator(engine, 'dbo_instrumentmaster')			
simulator = simulator.bu	uy_hold_sim()		
if simulator is not None:	or is not None:		
print('Incorrect return	value while running buy_hold_sim function: %s' %	simulator)	

# **APPENDIX B: KEY TERMS**

Term	Description
ARIMA	Auto-Regressive Integrated Moving Average
Back-End	Database that supports the middle layer and front-end
СМА	Cross Moving Average
EMA	Exponential Moving Average
FRL	Fibonacci Retracement Line
GMFSP_db	Name of the database used to store data
Instrument	Financial instruments are monetary contracts between parties. They can be created, traded, modified and settled. They can be cash, evidence of an ownership interest in an entity, or a contractual right to receive or deliver cash.
Middle-Layer	Python code that runs the algorithms, data fetch and, data loads
Non-Functional Requirement	A requirement that is deemed part of important and critical functions of a Software
Random Forest	An algorithm for making predictions on the data using the learning method of classification.
Tableau	Software used to visualize data
Test Case	Steps containing actions to test a functionality of the Software
Ticker	Stock symbol or instrument

THIS PAGE LEFT INTENTIONALLY BLANK