

Compliance Report

For

Transport Company Computerisation (TCC) Software

16.03.2022

Prepared by

CS VS

20CS10003 Abhinav Sen

20CS10012 Atulya Sharma

20CS30059 Yatharth Sameer

Department of Computer Science and Engineering, IIT Kharagpur

16 March 2022

Contents

1 Introduction and Important Points.....	2
2 Compliance of Unit testing of Modules.....	2
2.1 utility.py.	2
2.2 employee.py.	4
2.3 truck.py.	4
2.4 consign.py.	4
2.5 bill.py.....	4
2.5 mail.py.	5
2.6 manager.py.	5
2.7 main.py.	5
3 Testing integrated functionality and front end	9
3.1 Employee	9
3.1.1 Constructor	9
3.1.2 Login.	9
3.1.3 Create Consignment.	10
3.1.4 Forgot Password.	10
3.1.5 Deliver.....	10
3.1.6 Logout.....	10
3.2 Truck.	10
3.2.1 Constructor.	10
3.2.2 Change Driver	10
3.3 Consignment	11
3.3.1 Constructor.....	11
3.3.2 CalculateCost	11
3.4 Manager.	11
3.4.1 queryConsignment.	11
3.4.2 View Average waiting time of consignments.....	11
3.4.3 View average idle time period of trucks.....	11
3.4.4 query employee.	11
3.4.5 Query Branch.....	11
3.4.6 Query Truck.....	11
3.4.7 Add Employee.....	12
3.4.8 Remove Employee.....	12
3.4.9 View Truck Status.....	12
3.4.10 View truck usage.....	12
3.4.11 Change Truck Driver.....	12
3.4.12 Change rate.....	12
3.4.13 Add Branch.....	12
3.4.14 Add truck.....	12
3.4.15 Logout.....	12

1 Introduction and Important Points

This is the Compliance Report document for Transport company computerisation software. Here, we list

down test cases that were mentioned in the test suite along with the obtained PASS/FAIL results that were obtained after testing the software. note that for a positive test case, the expected result should be PASSED, while for a negative test case, the expected result should be FAILED.

2 Compliance of Unit testing of Modules

Unit testing for various modules:

Note: each individual function will be tested for correctness before moving on to the next module during unit testing. PASS indicates that the function is working as expected for the applied test cases

Overall application and front end testing is given in the next segment. This testing follows the test cases given in the test suite and involves testing functionality that is built upon the functions that have been unit tested

2.1 utility.py

functions:

- distance()
 - Either or both cities are non existent
 - FAIL
 - Cities are existent
 - PASS
- closestBranch ()
 - Address is not well formatted
 - FAIL
 - Address is well formatted
 - PASS
- checkLogin()
 - Either username or password is incorrect
 - FAIL
 - Both are correct and match in the database
 - PASS
- stringToDateTime()
 - Input is non string
 - FAIL
 - Input is string but not formatted
 - FAIL
 - Input is well formatted string

- PASS
- generateRandomString()
 - PASS
- deltaTimeToHours()
 - Either of the arguments is not in datetime type
 - FAIL
 - Both are of datetime or date type
 - PASS
- today()
 - PASS
- now()
 - PASS
- setupDB()
 - PASS
- mailPassword()
 - Wrong Email
 - PASS
 - Existent Email
 - PASS
- nextAvailableTruck()
 - Both arguments are of expected structured dictionary and string and exist in the database
 - PASS
 - Else
 - FAIL
- loadUnloadedConsignments()
 - Given branch does not exist
 - FAIL
 - Exists
 - PASS

Class: Branch

Functions:

- __init__
 - PASS
- convertToDictAndUpload
 - PASS
- convertFromDict()
 - Dictionary provided is of correct structure
 - PASS
 - Not of correct structure
 - FAIL

2.2 employee.py

Functions:

- changePassword() -PASS

Class: Employee

Functions:

- Employee__init__() --PASS
- convertToDictAndUpload() - -PASS

2.3 truck.py

Functions:

- changeDriver() -PASS
- dispatchTruck() -PASS
- unloadTruck() -PASS

Class: Truck

Functions:

- __init__() -PASS
- convertToDictAndUpload() -PASS

2.4 consign.py

Functions:

- dispatchConsignment()
 - Either of the IDs does not exist
 - FAIL
 - Both IDs exist
 - PASS
- loadConsignment()
 - ID of the consignment does not exist
 - FAIL
 - ID exists but given truck dictionary is None
 - PASS (handled)
 - Truck dictionary is wrong structured
 - FAIL
 - Truck is rightly structured and the ID exists in the database
 - PASS

Class: Consign

Functions:

- __init__()
 - PASS
- convertToDictAndUpload
 - PASS

2.5 bill.py

Functions:

- bill()

- Consignment is not of correct dictionary format
 - FAIL
- Consignment is of correct format
 - PASS

2.6 mail.py

Functions:

- sendMail()
 - Wrong / Non existent email ID
 - PASS (handled)
 - Existent ID
 - PASS

2.7 manager.py

Functions:

- branchQuery()
 - Branch exists
 - PASS
 - Does not exist
 - FAIL
- truckQuery()
 - Truck exists
 - PASS
 - Does not exist
 - FAIL
- employeeQuery()
 - PASS
- consignmentQuery()
 - PASS
- queryConsignmentsHeadedToSameBranch()
 - PASS
- calculateIdleTimeOfTruck()
 - Truck exists
 - PASS
 - Does not exist
 - FAIL
- viewTruckUsageInPeriod()
 - Truck exists
 - PASS
 - Does not exist
 - FAIL

2.8 main.py

Class: loginScreen

Functions:

- `__init__()`
 - PASS
- `login()`
 - PASS
- `accept()`
 - PASS
- `acceptSU()`
 - PASS
- `forgotpassword()`
 - PASS

Class: employeeScreen

Functions:

- `__init__()`
 - PASS
- `logout()`
 - PASS
- `enterDetailsPage()`
 - Any of the fields are empty
 - FAIL
 - Phone number is not a 10 digit integer
 - FAIL
 - Else
 - PASS
- `truckpage()`
 - PASS

Class: enterConsignmentDetailsScreen

Functions:

- `__init__()`
 - PASS
- `back()`
 - PASS
- `createConsignment()`
 - Volume is not a number
 - FAIL
 - Either Phone number is not a 10 digit integer
 - FAIL
 - Any field is empty
 - FAIL
 - Else
 - PASS

Class: truckScreen

Functions:

- `__init__()`
■ PASS
- `back()`
■ PASS
- `confirmArrivalOfTruck()`
■ PASS

Class: `managerScreen`

Functions:

- `__init__()`
■ PASS
- `goBack()`
■ PASS
- `goToViewEmployee()`
■ PASS
- `goToAddEmployee()`
■ PASS
- `goToViewTrucks()`
■ PASS
- `goToAddTruck()`
■ PASS
- `goToViewBranch()`
■ PASS
- `goToAddBranch()`
■ PASS
- `goToViewConsignment()`
■ PASS

Class: `ViewConsignmentScreen()`

Functions:

- `__init__()`
■ PASS
- `searchConsignment()`
■ PASS
- `back()`
■ PASS

Class: `ViewEmployeeScreen`

Functions:

- `__init__`
■ PASS
- `goBack()`
■ PASS
- `query()`
■ PASS
- `fireEmployee()`
■ PASS

Class: AddEmployeeScreen

Functions:

- `__init__()`
 - PASS
- `goBack()`
 - PASS
- `createEmployee()`
 - Phone number is not a 10 digit number
 - FAIL
 - Any of the field is empty
 - FAIL
 - Else
 - PASS

Class: ViewTrucksScreen

Functions:

- `__init__()`
 - PASS
- `goBack()`
 - PASS
- `query()`
 - PASS
- `history()`
 - Date is not formatted
 - FAIL
 - Date is formatted
 - PASS
- `calcIdleTime()` -PASS

Class: AddTruckScreen

Functions:

- `__init__()`
 - PASS
- `goBack()`
 - PASS
- `createTruck()`
 - Phone number of driver is not a 10 digit integer
 - FAIL
 - Any of the fields is empty
 - FAIL
 - Else
 - PASS

Class: ViewBranchScreen

Functions:

- `__init__()`
 - PASS

- goBack()
 - PASS
- query()
 - Branch does not exist
 - FAIL
 - Branch exists
 - PASS
- searchConsignmentHeaded()
 - PASS

Class: AddBranchScreen

Functions:

- __init__()
 - PASS
- goBack()
 - PASS
- createBranch()
 - Branch location already exists
 - FAIL
 - Any of the fields is empty
 - FAIL
 - Else
 - PASS

3. Testing integrated functionality and front end (Test Suite)

The following test cases follow from the test suite and build upon the integrated functions that have been unit tested in isolation. PASS indicates that the output of the functionality is as expected.

3.1 Employee

(including front end testing of the employee page)

1. Constructor-
 - Initialising an Employee Object with values as given in the test suite
 - works as expected, with fields clearly visible in the database
 - -PASS
2. Login -
 - if credentials match that of an existing employee, login is successful

- -PASS
- If either username or password doesn't match, login fails. Program raises an error message of invalid credentials
- -FAIL
- 3. Create Consignment
 - Entering valid details successfully creates a consignment to load. Once loaded, a message pops up with details regarding the same.
 - -PASS
 - In case the values provided are not valid, an error message pops up
 - -FAIL
 - If the volume of consignments to the same location greater than 500 and a truck is available for dispatch, dispatch occurs
 -
- 4. Forgot Password
 - If the forgot password button is clicked, the user gets an auto generated mail with a new randomly generated password. Works as expected
 - -PASS
- 5. Deliver
 - This functionality is used by the employee to confirm the arrival of a dispatched truck.
 - Upon being called, the truck object gets changed in the database as expected in the test suite
 - A bill for the customer is generated as required
 - Works as expected
 - -PASS
- 6. Logout
 - On clicking on the backButton, the user successfully logs out of the current page and moves back to the login page. Works as expected
 - -PASS

3.2 Truck

(constructor works as expected)

1. Constructor-
 - Initialising a Truck Object with values as given in the test suite works as expected, with fields clearly visible in the database
 - -PASS
 - In case of invalid field input, and error message pops up
 - -FAIL
2. changeDriver
 - Changing the driver with input as specified in the test suite works as expected, with changes clearly visible in the database
 - -PASS
 - Initialising incorrect values throws an error message
 - -FAIL

3.3 Consignment

1. Constructor-
 - Initialising a consignment with values as given in the test suite works as expected.
 - Upon entering incorrect values, an error message is thrown
 - -PASS
2. CalculateCost-
 - This method works as expected, with output decided algorithmically to calculate incurred charges for the consignment.
 - Updates revenue of the branch
 - -PASS

3.4 Manager

(including front end testing of manager page)

1. queryConsignment-
 - Query consignment works as detailed in the test
 - -PASS
 - If a wrong value is given as input, no output is shown
 - Output as expected
2. View average waiting time of consignments
 - This is shown in while querying a branch in a field
 - -PASS
 - Refer to 5th point
3. View average idle time period of trucks
 - This is shown in while querying a truck in a field
 - -PASS
 - Refer to 6th point
4. Query Employee
 - Upon entering employee name or id, employee details and statistics are displayed
 - If a wrong value is given as input, no output is shown
 - -PASS
5. Query Branch
 - Upon entering branch location, branch details and statistics are displayed
 - If a wrong value is given as input, no output is shown
 - -PASS
6. Query Truck
 - Upon entering Truck plateNo or id, truck details and statistics are displayed
 - If a wrong value is given as input, no output is shown
 - Works as expected
 - -PASS

7. Add Employee
 - Call the Employee constructor.
 - Works as expected
 - -PASS
8. Remove Employee
 - Input details of an employee to fire
 - The employee is removed from database as expected
 - -PASS
9. View truck status
 - Is covered in query truck
 - Refer to 6th point
 - -PASS
10. View truck usage
 - This is covered in query truck, and an button is made available to filter all delivery histories of a truck
 - If fields are put correctly and dates are formatted
 - -PASS
 - Otherwise
 - -FAIL (Handled exception)
11. Change Truck Driver
 - Input details of the new Truck Driver
 - In case fields are empty or the phone number is not a 10 digit integer (Handled exception)
 - -FAIL
 - Otherwise, works as expected
 - -PASS
12. Change rate
 - This is implemented in the manager screen
 - If the rate is a non number
 - -FAIL (Handled exception)
 - Else
 - -PASS
13. Add Branch
 - Upon entering Branch location, a new branch with no history is created and added to database
 - If an already registered branch is entered again, the program throws an error message displaying the same
 - Works as expected
 - -PASS
14. Add truck
 - Calls Truck Constructor
 - Works as expected
 - -PASS
15. Logout

- Logs the manager out of the application
- Works as expected
- -PASS