Testing:

Our program has gone under severe testing to ensure that it is bug and error free. To ensure that the program is bug free, we have implemented multiple testing listed in the table below

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type of test | Why | How | Result | Normal cases | Extreme cases |
| White Box Testing | To ensure that the input and output are correct | We have used multiple if else and print () statement in the program. We will have an expected output in our head and compare that output with the actual output. | If the result passes, then we continue, otherwise we will fix the code before continuing. | passed | N/A |
| Random Testing | To ensure that the functions will work with random inputs and outputs | test all functions and compare it with its output. Use random values as our input, formulate the output ourselves and compare the result using junits. | If we fail something in junits, then we try to figure out why. If we pass it, then we continue with the remaining work | Passed | Passed |
| Partition Testing | To ensure that the functions will work with any kind of inputs and outputs that represent different kind of inputs. | We use values like “Kraaifontein to Brackenfell on Monday”, if this works, then all trains in that partition (Area North with only 1 train) should work. You can see out inputs in testing.py | If one testing fails, we assume that all partition related to that test fails. So we fix the problem related to that partition | Passed | Passed |
| Acceptance test | To see whether the user will be able to use the program to their satisfaction | We have manually checked whether the output created is correct and able to use. We see if the shortest time is the shortest if the closest time to depart is the closest time. | When we found something that we do not like about the output, we tried to change our code so that the output is acceptable | Passed | Passed |
| Use-case based black box: | To ensure the customers satisfactory | We have run the website on different computers to ensure that it works with them. We have asked multiple people to use the application to hear from their feedback. | The website functions just fine with different computers. From feedbacks, We improve our website for better user experience | Passed | N/A |
| Recovery test | To ensure that data is not lost due to break down or accidental close of the application | We do many different testings, once bug is found, we fix them immediately. We tried to store then close website immediately, check if the database was able to record the data. | The program is working better every time we fix a bug. Everything is recorded in the database, so nothing is lost when website is closed suddenly after the clicking of the button | Passed | Passed |
| Security test | To ensure the privacy of the users | we have created a user and password protection to ensure that all users have their own account if they want it. We tried to log in with different accounting using different passwords. Check if the user is able to log in. | When the user an username with wrong password, there will be an error preventing them from accessing the other account’s data | Passed | N/A |
| Performance test | To improve efficiency | We import time.  User time.start() and time.time() to measure the time taken to execute different inputs | Simon’s Town – Strand on Monday took 0.006  Mowbray – Stellenbosch on Saturday took 0.005  Cape Town to Avondale on Monday took 0.004.  The time normally won’t go over 0.01 seconds | Passed | Passed |
| Spelling Testing | To ensure that all spellings of the train stations are correct |  |  |  |  |

