**UAT Test Procedure**

For each of the tests below, run the test according to the test procedure and test inputs. Observe the output. If it matches all the expected results, the test passes. Otherwise, it fails. Note down the actual output in the “Actual Results” field.

Download the test data from <http://ares.smu.edu.sg/uat/uat-data-2019.zip>. Inside you will find two data files, uat-data1.zip and uat-data2.zip. Use uat-data1.zip for test cases 1 to 23 and uat-data2.zip for test case 24. Public IP address is <http://202.161.45.182/uat/uat-data-2019.zip>

|  |  |
| --- | --- |
| **Group (GX-TY):** |  |
| **Tester:** |  |
| **Timestamp of deployment** |  |
| **IP address of instance** |  |
| **Number of tests passed** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S/N** | **Test Case** | **Test Inputs** | **Test Procedure** | **Expected Results** | **Actual Results**  **(Pass / Fail)** |
| **0** | **App is deployed**  **Correctly** | IP address of the app | 1. Access <http://ip_address/app> | 1. Login page should be shown. 2. If nothing shows or you see a directory listing, this fails and the entire UAT is a 0. |  |
| **1** | **Upload**  Test if Bootstrap works | uat-data1.zip  **Admin id/pwd:** <Provided by team> | 1. Login to admin interface. 2. Upload the uat-data1.zip. | 1. A message that the data is uploaded correctly (there should be no error messages printed).  |  |  | | --- | --- | | File | # of Fields | | bid.csv | 11 | | course.csv | 13 | | course\_completed.csv | 5 | | prerequisite.csv | 2 | | section.csv | 57 | | student.csv | 14 |  1. Round 1 is started automatically.   **Note**: Having to change the bootstrap data filename is considered a failure of this test case.  **Note 2**: Output should not be in JSON format either |  |
| **2** | **Login** Validate that a student with incorrect credentials cannot log in | **Username:**  amy.ng.2009  **Password:** qwerty128 | 1. Enter the EXACT username and password. 2. Press submit | 1. “Username/password incorrect” (or a similar message) should be displayed. |  |
| **3** | **Login**  Validate that a student with correct credentials can log in | **Username:**  fluffy  **Password:** 6789 | 1. Enter the username and password. 2. Press submit. | 1. The home page should be displayed. 2. The e$ account is left with $50.00. |  |
| **4** | **e$ account mgmt** Validate the correct amount is deducted when processing bid.csv. | **Username:**  evil  **Password:** 5678 | 1. Login as evil. | 1. The home page should be displayed.   The e$ account is left with $35.00.  **Note**: If this test case fails, manually add the following bid to evil. This is needed for the later test case.  **userid**: evil  **amount**: 15  **course**: T1234YZ  **section**: S1 |  |
| **5** | **Add Bid** Validate that courses from the School of Computer Science (SCS) are not available for funny (who is attending SMOO). | **Username:**  funny  **Password:** 6789 | 1. Login with the given username and password. 2. Try to bid for the SCS courses. | Students should not be able to view/bid for the SCS courses.  OR  System does pre-filtering (e.g., it shows only SMOO courses for bidding). |  |
| **6** | **Add Bid** Validate that a student cannot bid for a course when the pre-requisite is not fulfilled. | **Username:**  heaven  **Password:** 8901  **Course:** IS002  **Section:** S1  **Bid:** $10 | 1. Login as heaven. 2. Add the section with bid amount $10. | System should display an error message that the pre-requisite is not fulfilled.  OR  System does pre-filtering (i.e., this section is not listed). |  |
| **7** | **Add Bid** Validate that a student can bid for a course when the pre-requisite is fulfilled. | **Username:**  evil  **Password:** 5678  **Course:** IS002  **Section:** S1  **Bid:** $24.99 | 1. Login as evil. 2. Add the section (IS002 S1) with bid amount $24.99. | 1. Section is displayed as bid for. 2. The e$ account is left with $10.01. |  |
| **8** | **Add Bid**  Validate that a student cannot bid for a course when the exam falls on the same day and time | **Username:**  evil  **Password:** 5678  **Course:** T1234PZ  **Section:** S1  **Bid:** $10 | 1. Login as evil. 2. Add the section with bid amount $10. | The student should not be able to bid.  OR  System does pre-filtering (i.e., this section is not listed). |  |
| **9** | **Add Bid**  Validate that a student can bid for a course when the exam falls on the same day but different timing and check e$ handling | **Username:**  evil  **Password:** 5678  **Course:** T1234EZ  **Section:** S2  **Bid:** $10 | 1. Add the section with bid amount $10. | 1. Section is displayed as having being bid for. 2. The e$ account is left with $0.01. |  |
| **10** | **Drop Bid**  Check if e$ handling is correct | **Username:** evil  **Password:** 5678  **Course:** T1234EZ  **Section:** S2 | Drop the bid specified to the left.  **Note**: if the previous test case fails, add any section with bid amount of $10. Use the added bid for this test case. | 1. Section should be dropped successfully. 2. Account should be left with $10.01 |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **11** | **Show Bidding Results** | **username:**  benny  **password:** 2345 | 1. Login in as benny. 2. Check the bidding results. | 1. The following results should be displayed.  |  | | --- | | IS004, S3, 10.01,  Pending | | T1234EZ, S4, 10.01,  Pending | | T1234PZ, S2, 10.01,  Pending | | T1234YZ, S9, 10.01,  Pending | | JUMPING, S99, 10.01,  Pending |   \* The table should include course id, section number, bid amount, status of bidding.   1. His account should be left with $4.95. |  |
| **12** | **Check bid.csv processing is correct.** (tie bids at the clearing price and overriding of subsequent bids) | **Student’s Login/pwd:**  chloe/3456 danny/4567  funny/6789 | Login in to chloe, danny, and funny’s account. | 1. Check that the bids for chloe and danny (IS003, section S1) are pending. 2. Check that the e$ balance of both chloe and danny is $39. 3. Check that funny has a bid for IS003 section S2 (pending) and e$ balance of $10.01. |  |
| **13** | **Check e$ balance is handling small amounts correctly** | **Student’s Login/pwd:**  funny/6789  **Course:**  IS004  **Section:**  S5  **Bid:** $10.01 | Login in to funny’s account and bid for IS004-section S5 with a value of $10.01. | 1. Check that IS004 has been added and that funny’s e$ balance is $0.00. |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **14** | **Stop Round 1** | **Admin id/pwd:** <Provided by team> | 1. Login in as admin. 2. Close Round 1. | A message should show that the round is stopped and cleared successfully. |  |
| **15** | **Round 1 Clearing Logic** (tie bids at the clearing price) | **Student’s Login/pwd:**  chloe/3456 danny/4567 | Login in to chloe and danny's account. | Check the following:   1. Round 1 is stopped. No bids can be added. 2. The bidding results show that the IS003 bid has failed for both chloe and danny. 3. Check that both accounts have $50 ($11 must be refunded back to them).   **Note**: ALL checks must pass to pass this test case. |  |
| **16** | **Start Round 2.** | **Admin id/pwd:** <Provided by team>  **Username:**  evil  **Password:** 5678 | 1. Login as admin. 2. Start round 2. 3. Login in as evil. | 1. Check that she was successful with her previous bids.  |  |  | | --- | --- | | **Course:** | T1234YZ | | **Section:** | S1 | | **Bid:** | $15.00 |  |  |  | | --- | --- | | **Course:** | IS002 | | **Section:** | S1 | | **Bid:** | $24.99 |  1. Her account should be left with $10.01. 2. Validate that round 2 has started. Verify that Evil can bid for SCS modules now; she is attending SMOO.   Note: please delete any bid you add |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **17** | **Round 2 bidding** Checks that the minimum bid value is 10 when  Num(size) > num(bids) | **Username:**  gabriel  **Password:** 7890 | 1. Login in as Gabriel. 2. **DO NOT MAKE ANY BID!** | Check the following   1. IS002 S1 has 1 vacancy and a current minimum bid price of $10. | |  |
| **18** | **Add Round 2 bid**  Validate that the system accepts bids during round 2 | **Username:**  gabriel  **Password:**  7890  **Course**  IS002  **Section:** S1  **Amount:** 10 | 1. Add the section with a bid amount of $10. | 1. Check that gabriel's account has $40 left and IS002(S1) is added to his bid results. 2. Check that the minimum bid price for IS002 S1 is $11. 3. Check that the minimum bid price for IS002 S3 is still $10. 4. Check that IS002 S1 still has 1 vacancy (note that round 2 is not cleared). |  | |
|  |
| **19** | **Round 2 Bidding Logic** (bid value starts at 10 and never goes down even if a vacancy is released back to the pool) | **Username:**  evil  **Password:** 5678  **Course**  IS002  **Section:** S1 | 1. Login in as evil. 2. Drop IS002 S1. | 1. Check that the minimum bid value for IS002 S1 is still $11. 2. Check that the minimum bid price for IS002 S3 is still $10. 3. e$ balance for evil should be $35. 4. Check that IS002 S1 now has 2 vacancies. |  | |
| **20** | **Round 2 Bidding Logic**  Check that a student can bid for a vacancy that has been returned back to the pool | **Username:**  fluffy  **Password:**  6789  **Course**  IS002  **Section:** S1  **Amount:** 11 | This should be done by tester 2 on another computer (or a different browser on the same laptop).   1. Login as fluffy. 2. Bid for the section   **(DO NOT LOGOUT.)** | Check for:   1. The system should add the bid to the student's schedule. 2. The minimum bid value for IS002 S1 should be $11 after fluffy’s bid. 3. Check that the minimum bid price for IS002 S3 is still $10. 4. fluffy’s e$ balance is $39 5. Check that IS002 S1 has 2 vacancies. |  | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **21** | **Round 2 Bidding Logic** (real-time losing of bid) | **Username:**  evil  **Password:** 5678  **Course**  IS002  **Section:** S1  **Amount:** 12 | **Part 1: Done on laptop 1 using evil**   1. Login in as evil. 2. Bid for the section. | **Part 1:**  Check the following before bidding   1. The minimum bid value for IS002 S1 should be $11.   Check the following after bidding for IS002 S1   1. Evil’s e$ balance is $23. 2. The minimum bid value should be $12. 3. IS002 S1 has 2 vacancies. |  |
| **Part 2: Done on laptop 2 using fluffy**  (continue from previous test case)   1. On fluffy's bid status page (on the other laptop), refresh the page. 2. Log out from fluffy's account. | **Part 2:**  Check the following on fluffy's screen   1. The minimum bid value for IS002 S1 should be $12. 2. fluffy still has a successful bid for the section. 3. e$ balance of $39. 4. IS002 S1 has 2 vacancies. |
| 1. Log in as gabriel(password: 7890). | Check the following on gabriel's screen   1. The minimum bid value for IS002 S1 should be $12. 2. gabriel has lost his bid for the IS002 S1. 3. gabriel’s e$ balance should still be $40 4. IS002 S1 has 2 vacancies. |
| **22** | **Round 2 Handling of Minimum Clearing Price**  (check that price never goes down) | **Username:**  apple  **Password:** 1234  **Course**  CS001  **Section:** S1  **Amount:** 10 | 1. Add the section with a bid value of 10 for apple | 1. apple should have CS001 added 2. e$ balance of $40.00. 3. minimum bid price for CS001 S1 should be 10 4. CS001 S1 has 2 vacancies. |  |
| 1. Login as busybody (pwd: 2345) and add CS001 S1 with a bid value of 12 | 1. busybody should have CS001 added 2. e$ balance of $38.00. 3. minimum bid price for CS001 S1 should be 11 4. CS001 S1 has 2 vacancies. |  |
| 1. Login as charlie (pwd: 3456) and add CS001 S1 with a bid value of 301 | 1. charlie should have CS001 added 2. e$ balance of $49.00. 3. minimum bid price for CS001 S1 should be 13 4. apple should lose their bid and have e$ of 40 5. CS001 S1 has 2 vacancies. |
| 1. Login as desperate (pwd: 4567) and add CS001 S1 with a bid value of 350 | 1. desperate should have CS001 added 2. e$ balance of $0.00. 3. minimum bid price for CS001 S1 should be 302 4. busybody should lose their bid and have e$ of 38 5. CS001 S1 has 2 vacancies. |
| 1. Login as charlie and drop CS001 S1 | 1. charlie should have e$ of 350 and no bid for CS001 S1 2. busybody should get back their bid for CS001 S1 and their e$ should be $38. 3. apple should still lose their bid for CS001 with e$ of 40 4. minimum bid for CS001 S1 should still be 302 5. CS001 S1 has 2 vacancies. |
| 1. Login as desperate and drop CS001 S1 | 1. desperate should have e$ of 350 and no bid for CS001 S1 2. apple should get back their bid for CS001 S1 and their e$ should be $40. 3. busybody should still have their bid for CS001 with e$ of 38 4. minimum bid for CS001 S1 should still be 302 5. CS001 S1 has 2 vacancies |
| **23** | **Round 2 Clearing Logic** (tie bids at the clearing price) | **Student’s Login/pwd:**  chloe/3456 danny/4567  **Admin id/pwd:** <Provided by team> | 1. Login as chloe. 2. Add a bid for IS004(S4) for $30.00. 3. Login as danny. 4. Add a bid for IS004(S4) for $30.00. 5. Login as admin. 6. Close Round 2. | danny & chloe’s bids for IS004(S4) should pass.   1. Check that both accounts (danny and chloe) have e$ = $20. 2. Both should have successful bids for IS004(S4) . 3. Check that evil and fluffy have successful bids for IS002 S1. e$ of $23 (evil) and $39 (fluffy). 4. Gabriel does not have IS002 S1 and a e$ balance of 50 5. Check that apple (e$ 40) and busybody (e$ 38) have successful bids for CS001 S1. 6. charlie and desperate have no sections and e$ = 350 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **24** | **Bootstrap validation**  Validate that the system checks for invalid bids | **Admin id/pwd:** <Provided by team> | 1. Login as admin. 2. Upload uat-data2.zip.   **NOTES For This Test:**   1. The admin output must be human readable and not in JSON format as stated in the project writeup. If the error messages are a JSON format, this test fails. 2. The errors should be displayed (with proper error messages):   **Note**: The ordering of the errors is not crucial for Web UI except that it must be clear which file the error being reported is for.  **However**, it must be clear which record in which file has the errors reported. The best reports will state the file, line number, and the specific error. If it is not clear which specific file or record corresponds to the error, this test fails. | The errors are:  **Bid.csv (7 errors)**   1. Line 2:   - Invalid amount (amy, 9.99, IS003, S1)   1. Line 3: - Invalid section (benny, IS003, S6) 2. Line 4:   - Invalid userid (truck does not exist in student.csv)   1. Line 5: - course completed - not own school course (evil, IS001, S2) 2. Line 7:   - class timetable clash (fluffy, IS003, S4); conflicted with (fluffy, IS002, S2)   1. Line 13:   - section limit reached (elene, IS008, S2)   1. Line 14: - not enough e-dollar (fluffy, 70, IS003, S5)   **Courses.csv (3 errors)**   1. line 4   - Invalid exam start (CS003, 900)   1. line 10   - Invalid exam date (IS004, abcdefg)   1. line 11   - Invalid exam date (IS005, 201012232)  **Course\_completed.csv (3 errors)**   1. Line 2:   - Invalid userid (fun)   1. Line 3:   - Invalid course (IS001a)   1. Line 4:   - Invalid course completed (Gabriel, IS002. No pre-req)  **Prerequisite.csv (2 errors)**   1. Line 2:   Invalid course (IS0021)   1. Line 3:   Invalid prerequisite (CS001a)  **Section.csv (12 errors)**   1. line 2:   -Invalid course (JUMPING not found in course.csv)   1. line 4:   -Invalid section (IS001, T5)   1. line 5:   -Invalid day (IS001, S4, 8)   1. line 7:   -Invalid day (IS001, S6, Monday)   1. Line 8:   -Invalid start  (IS002, S1, 800)   1. Line 14:   -Invalid size (IS003, S2, 0)   1. Line 15:   -Invalid size (IS003, S3, r)   1. Lines 18, 19, 20, 21, 22:   - Invalid course (IS004 rejected for invalid exam date)  **Student.csv (3 errors)**   1. Line 9:   Duplicate userid (evil koh, evil ‘s twin brother baby evil)   1. Line 10:   -Invalid edollar (apple, 123.34.34)   1. Line 11:   -Invalid edollar (busy body, abc) |  |

Please provide feedback to the team about their User Interface. In particular, let them know

1. What you like about the User Interface
2. What you don’t like about the User Interface

|  |
| --- |
| **Do note that Usability feedback is usually not part of the User Acceptance Test.** |