CSC-204 Software Engineering

Report

Project Testing

Submitted by:

Abdul Muiz	2024-SE-14
Umair Arshad	2024-SE-38
Usman Shahid	2024-SE-35

Submitted to:

Sir Muzammil

Dated: 4th January 2026

Department of Computer Science
University of Engineering and Technology Lahore, New Campus

Table of Contents

Software Testing Documentation - Online Voting System

SYSTEM LEVEL TEST CASES

Tester Name: Umair

Test Case ID: SYS_001

Test Case Description: Test End-to-End Voter Registration and Voting Flow

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: October 10, 2025
Test Case Status: Pass

Prerequisites:

1. Database connection established
2. MySQL server running
3. Voting system application compiled and running

Test Data:

- CNIC: 3520212345678
- Name: Ahmed Khan
- Province: Punjab
- District: Lahore
- Password: voter123

Test Scenario:

Verify that a new voter can register, login, and cast votes for both MNA and MPA when elections are active.

Step #	Step Details	Expected Results	Actual Results	Status
1	Launch application and select option 3 (Voter)	Voter portal menu displayed	As Expected	Pass
2	Select Sign Up option	Registration form displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
3	Enter valid CNIC (13 digits)	CNIC accepted	As Expected	Pass
4	Enter name, province, district, password	All fields accepted	As Expected	Pass
5	Complete registration	Success message displayed	As Expected	Pass
6	Login with registered credentials	Login successful, voter menu displayed	As Expected	Pass
7	Admin starts National election	MNA voting option becomes available	As Expected	Pass
8	Cast vote for MNA candidate	Vote recorded successfully	As Expected	Pass
9	Admin starts Local election for Punjab	MPA voting option becomes available	As Expected	Pass
10	Cast vote for MPA candidate	Vote recorded, thank you message shown	As Expected	Pass

---

Test Case ID: SYS_002

Test Case Description: Test Duplicate CNIC Registration Prevention

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: October 12, 2025
Test Case Status: Fail

Prerequisites:

1. Database contains existing voter with CNIC 3520212345678
2. Application running

Test Data:

- CNIC: 3520212345678 (already exists)
- Name: Ali Hassan
- Province: Sindh
- Password: test456

Test Scenario:

Verify that system prevents registration with duplicate CNIC.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to Voter Sign Up	Registration form displayed	As Expected	Pass
2	Enter existing CNIC: 3520212345678	System should check CNIC	System accepts input	Pass
3	Enter name and other details	Fields accepted	As Expected	Pass
4	Submit registration	Error: "CNIC already registered"	Registration succeeded (BUG)	**Fail**

Bug Report: System allows duplicate CNIC registration. Data integrity issue found.

Test Case ID: SYS_003

Test Case Description: Test Duplicate CNIC Registration Prevention (Fixed)

Created By: Umair
Reviewed By: Usman
Version: 2.0
Date Tested: October 15, 2025
Test Case Status: Pass

Prerequisites:

1. Bug fix applied to Voter.h registerVoter() function
2. Database contains existing voter with CNIC 3520212345678
3. Application recompiled

Test Data:

- CNIC: 3520212345678 (already exists)
- Name: Sara Ahmed
- Province: KPK

Test Scenario:

Verify that duplicate CNIC registration is now properly prevented.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to Voter Sign Up	Registration form displayed	As Expected	Pass
2	Enter existing CNIC: 3520212345678	System checks database	As Expected	Pass
3	Enter remaining details	Fields accepted	As Expected	Pass
4	Submit registration	Error message: "This CNIC is already registered"	As Expected	Pass
5	Verify no duplicate entry in database	No new record created	As Expected	Pass

---

Test Case ID: SYS_004

Test Case Description: Test Admin Election Management Complete Workflow

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: October 18, 2025
Test Case Status: Pass

Prerequisites:

1. Admin credentials available (admin/admin123)
2. At least 5 candidates in database (3 MNA, 2 MPA)
3. At least 3 registered voters

Test Data:

- Admin Username: admin
- Admin Password: admin123
- Province: Punjab

Test Scenario:

Verify complete election lifecycle from admin perspective: start elections, voters vote, stop elections, view results.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as Admin	Admin menu displayed	As Expected	Pass
2	Select "Start Voting" → National Election	Election status updated to 'start'	As Expected	Pass
3	Select "Start Voting" → Local Election for Punjab	Local election started	As Expected	Pass
4	Voters cast their votes (simulate 3 voters)	All votes recorded	As Expected	Pass
5	Select "Stop Voting" → National Election	National election stopped	As Expected	Pass
6	Select "Stop Voting" → Local Election	Local election stopped	As Expected	Pass
7	Select "View Results" → National	Results displayed with vote counts	As Expected	Pass
8	Select "View Results" → Local	Results displayed, Chief Minister announced	As Expected	Pass

---

Test Case ID: SYS_005

Test Case Description: Test Invalid CNIC Format Validation

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: October 20, 2025
Test Case Status: Pass

Prerequisites:

1. Application running
2. Access to voter registration

Test Data:

- CNIC Variations:
  - 123 (too short)
  - 12345678901234 (too long)

- 1234567890ABC (contains letters)
- 12345-6789012-3 (contains dashes)

Test Scenario:

Verify that system validates CNIC format correctly.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to voter registration	Form displayed	As Expected	Pass
2	Enter CNIC: 123	Error: Invalid format	As Expected	Pass
3	Enter CNIC: 12345678901234	Error: Invalid format	As Expected	Pass
4	Enter CNIC: 1234567890ABC	Error: Invalid format	As Expected	Pass
5	Enter CNIC: 12345-6789012-3	Error: Invalid format	As Expected	Pass
6	Enter valid CNIC: 3520212345678	CNIC accepted	As Expected	Pass

---

Test Case ID: SYS_006

Test Case Description: Test Voter Cannot Vote Twice in Same Election

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: October 22, 2025
Test Case Status: Pass

Prerequisites:

1. Voter registered and logged in
2. National election active
3. Voter has already cast vote for MNA

Test Data:

- CNIC: 3520212345678
- Password: voter123

Test Scenario:

Verify that a voter cannot cast multiple votes in the same election type.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as voter	Voter menu displayed	As Expected	Pass
2	Vote for MNA candidate (first time)	Vote cast successfully	As Expected	Pass
3	Voter menu reloaded	"Vote for MNA" option should be disabled/hidden	As Expected	Pass
4	Attempt to vote for MNA again	Message: "Cannot vote for MNA at this time"	As Expected	Pass
5	Check database voter record	has_voted_mna = 1	As Expected	Pass

---

Test Case ID: SYS_007

Test Case Description: Test Candidate Login and View Details

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: October 25, 2025
Test Case Status: Fail

Prerequisites:

1. Candidate exists in database
2. Candidate ID: 5, Password: cand123

Test Data:

- Candidate ID: 5
- Password: cand123
- Type: MNA

Test Scenario:

Verify candidate can login and view their details and vote count.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select Login as Candidate	Candidate type selection displayed	As Expected	Pass
2	Select MNA (option 1)	Login prompt shown	As Expected	Pass
3	Enter Candidate ID: 5	ID accepted	As Expected	Pass
4	Enter Password: cand123	Login attempt made	Error: no password field in DB	**Fail**

Bug Report: Candidates table missing 'password' column. Cannot authenticate candidates.

Test Case ID: SYS_008

Test Case Description: Test Candidate Login and View Details (Fixed)

Created By: Umair
Reviewed By: Usman
Version: 2.0
Date Tested: October 28, 2025
Test Case Status: Pass

Prerequisites:

1. Database schema updated with password column
2. Candidate ID: 5, Password: cand123
3. Candidate has received some votes

Test Data:

- Candidate ID: 5
- Password: cand123

Test Scenario:

Verify candidate authentication and details viewing after fix.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select Login as Candidate → MNA	Login form displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
2	Enter ID: 5, Password: cand123	Login successful	As Expected	Pass
3	Select "View My Details"	Name, party, province, district, age displayed	As Expected	Pass
4	Select "View My Votes"	Current vote count displayed	As Expected	Pass
5	Logout	Returned to main menu	As Expected	Pass

Test Case ID: SYS_009

Test Case Description: Test Election Timing Control

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: October 30, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Elections initially stopped

Test Data:

- Province: Sindh

Test Scenario:

Verify voters can only vote when elections are active.

Step #	Step Details	Expected Results	Actual Results	Status
1	Voter logs in (election not started)	Voter menu shown	As Expected	Pass
2	Check available voting options	No MNA/MPA voting options available	As Expected	Pass
3	Admin starts National election	Election status changed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
4	Voter refreshes menu	MNA voting option now available	As Expected	Pass
5	Cast MNA vote	Vote successful	As Expected	Pass
6	Admin stops National election	Election status changed	As Expected	Pass
7	New voter attempts MNA vote	Message: Cannot vote at this time	As Expected	Pass

---

Test Case ID: SYS_010

Test Case Description: Test Admin Add Candidate Functionality

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: November 2, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Database accessible

Test Data:

- Name: Zainab Ali
- Party: PartyA
- Type: MNA
- Province: Balochistan
- District: Quetta
- Age: 45

Test Scenario:

Verify admin can successfully add new candidates to the system.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select "Add Candidate" from admin menu	Add candidate form displayed	As Expected	Pass
2	Enter candidate name: Zainab Ali	Name accepted	As Expected	Pass
3	Enter party: PartyA	Party accepted	As Expected	Pass
4	Enter type: MNA	Type accepted	As Expected	Pass
5	Enter province: Balochistan	Province accepted	As Expected	Pass
6	Enter district: Quetta	District accepted	As Expected	Pass
7	Enter age: 45	Age accepted	As Expected	Pass
8	Submit form	Success message displayed	As Expected	Pass
9	Verify in database	New candidate record exists with votes=0	As Expected	Pass

---

Test Case ID: SYS_011

Test Case Description: Test Admin Remove Candidate Functionality

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: November 5, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Candidate with ID=10 exists in database

Test Data:

- Candidate ID to remove: 10

Test Scenario:

Verify admin can remove candidates from the system.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select "Remove Candidate"	Prompt for candidate ID shown	As Expected	Pass
2	Enter candidate ID: 10	ID accepted	As Expected	Pass
3	Confirm removal	Success message displayed	As Expected	Pass
4	Verify in database	Candidate ID 10 no longer exists	As Expected	Pass
5	Attempt to remove non-existent ID: 999	Error message shown	As Expected	Pass

---

Test Case ID: SYS_012

Test Case Description: Test Results Display for Voters

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: November 8, 2025
Test Case Status: Pass

Prerequisites:

1. Voter logged in (Province: Punjab, District: Lahore)
2. Multiple candidates with various vote counts exist
3. Some votes have been cast

Test Data:

- Voter CNIC: 3520212345678
- Province: Punjab
- District: Lahore

Test Scenario:

Verify voters can view election results for their region.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as voter	Voter menu displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
2	Select "View Results"	Results page displayed	As Expected	Pass
3	Check results content	Shows candidates from Punjab/Lahore	As Expected	Pass
4	Verify vote counts	Vote numbers displayed correctly	As Expected	Pass
5	Check sorting	Results sorted by votes descending	As Expected	Pass
6	Verify both MNA and MPA shown	Both types visible	As Expected	Pass

---

Test Case ID: SYS_013

Test Case Description: Test Invalid Input Handling in Menus

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: November 10, 2025
Test Case Status: Pass

Prerequisites:

1. Application running
2. Main menu displayed

Test Data:

- Invalid inputs: abc, @#$, 999, -1, empty input

Test Scenario:

Verify system handles invalid menu inputs gracefully without crashing.

Step #	Step Details	Expected Results	Actual Results	Status
1	At main menu, enter "abc"	Error message, menu redisplayed	As Expected	Pass
2	Enter special characters "@#$"	Error message, menu redisplayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
3	Enter out of range: 999	Invalid choice message	As Expected	Pass
4	Enter negative number: -1	Invalid choice message	As Expected	Pass
5	Enter valid option: 1	Admin login screen shown	As Expected	Pass

---

Test Case ID: SYS_014

Test Case Description: Test Database Connection Failure Handling

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: November 12, 2025
Test Case Status: Fail

Prerequisites:

1. MySQL server stopped
2. Application ready to launch

Test Data:

- None

Test Scenario:

Verify system handles database connection failure gracefully.

Step #	Step Details	Expected Results	Actual Results	Status
1	Stop MySQL server	Server not running	As Expected	Pass
2	Launch application	Error message about connection failure	Application crashes	**Fail**
3	Application should not crash	Graceful error and exit	Application terminated abruptly	**Fail**

Bug Report: Application crashes when database is unavailable. Need better error handling.

Test Case ID: SYS_015

Test Case Description: Test Database Connection Failure Handling (Fixed)

Created By: Umair
Reviewed By: Usman
Version: 2.0
Date Tested: November 15, 2025
Test Case Status: Pass

Prerequisites:

1. Error handling code added to DatabaseManager::connect()
2. MySQL server stopped

Test Data:

- None

Test Scenario:

Verify improved database connection error handling.

Step #	Step Details	Expected Results	Actual Results	Status
1	Stop MySQL server	Server not running	As Expected	Pass
2	Launch application	Clear error message displayed	As Expected	Pass
3	Check application behavior	Application exits gracefully with exit code	As Expected	Pass
4	Start MySQL server	Server running	As Expected	Pass
5	Launch application again	Successful connection message	As Expected	Pass

---

Test Case ID: SYS_016

Test Case Description: Test Voter Regional Filtering

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: November 18, 2025
Test Case Status: Pass

Prerequisites:

1. Multiple candidates from different provinces/districts in database
2. Voter from Punjab, Lahore

Test Data:

- Voter Province: Punjab
- Voter District: Lahore
- MNA candidates: 3 from Punjab, 2 from Sindh
- MPA candidates: 2 from Lahore, 1 from Multan

Test Scenario:

Verify voters only see candidates from their region.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as voter (Punjab/Lahore)	Login successful	As Expected	Pass
2	Start National election	Election active	As Expected	Pass
3	Select "Vote for MNA"	List of MNA candidates shown	As Expected	Pass
4	Verify candidates shown	Only Punjab MNA candidates visible	As Expected	Pass
5	Start Local election for Punjab	Election active	As Expected	Pass
6	Select "Vote for MPA"	List of MPA candidates shown	As Expected	Pass
7	Verify candidates shown	Only Lahore MPA candidates visible	As Expected	Pass

---

Test Case ID: SYS_017

Test Case Description: Test Prime Minister Selection Logic

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: November 20, 2025
Test Case Status: Pass

Prerequisites:

1. National elections completed in all provinces
2. Vote counts recorded

3. Admin viewing national results

Test Data:

- Punjab winner: PartyA
- Sindh winner: PartyA
- KPK winner: PartyB
- Balochistan winner: PartyA

Test Scenario:

Verify Prime Minister is selected based on provincial wins.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin stops national elections	Elections stopped	As Expected	Pass
2	Select "View Results" → National	Results page displayed	As Expected	Pass
3	System counts provincial wins	PartyA: 3, PartyB: 1	As Expected	Pass
4	PM announcement displayed	"Prime Minister from PartyA"	As Expected	Pass

---

Test Case ID: SYS_018

Test Case Description: Test Chief Minister Selection Logic

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: November 22, 2025
Test Case Status: Pass

Prerequisites:

1. Local election completed in Punjab
2. MPA votes recorded

Test Data:

- Province: Punjab
- Highest MPA votes: Candidate from PartyA

Test Scenario:

Verify Chief Minister is selected based on highest MPA votes.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin stops local election for Punjab	Election stopped	As Expected	Pass
2	Select "View Results" → Local → Punjab	Results displayed	As Expected	Pass
3	System identifies highest votes	Correct candidate identified	As Expected	Pass
4	CM announcement displayed	"Chief Minister from party: PartyA"	As Expected	Pass

---

Test Case ID: SYS_019

Test Case Description: Test Concurrent User Scenarios

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: November 25, 2025
Test Case Status: Pass

Prerequisites:

1. Multiple application instances can run
2. National election active
3. Same candidate available

Test Data:

- Candidate ID: 5
- Initial votes: 10

Test Scenario:

Verify system handles multiple voters voting simultaneously.

Step #	Step Details	Expected Results	Actual Results	Status
1	Launch 3 application instances	All running simultaneously	As Expected	Pass
2	Login 3 different voters	All logged in	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
3	All vote for same candidate (ID 5)	Votes processing	As Expected	Pass
4	Check final vote count	Should be 13 (10 + 3)	As Expected	Pass
5	Verify no vote loss	All votes recorded	As Expected	Pass

Test Case ID: SYS_020

Test Case Description: Test Admin Invalid Credentials

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: November 28, 2025
Test Case Status: Pass

Prerequisites:

1. Application running
2. Main menu displayed

Test Data:

- Invalid username: wrongadmin
- Invalid password: wrongpass
- Valid username: admin
- Valid password: admin123

Test Scenario:

Verify admin authentication security.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select "Login as Admin"	Login prompt shown	As Expected	Pass
2	Enter username: wrongadmin	Username accepted for input	As Expected	Pass
3	Enter password: wrongpass	Password accepted for input	As Expected	Pass
4	Submit credentials	Error: Invalid admin credentials	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
5	Returned to main menu	Main menu displayed	As Expected	Pass
6	Try again with correct credentials	Login successful	As Expected	Pass

---

Test Case ID: SYS_021

Test Case Description: Test Voter Login Invalid Credentials

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: December 1, 2025
Test Case Status: Pass

Prerequisites:

1. Registered voter exists (CNIC: 3520212345678, Password: voter123)

Test Data:

- Wrong CNIC: 1111111111111
- Wrong Password: wrongpass
- Correct CNIC: 3520212345678
- Correct Password: voter123

Test Scenario:

Verify voter authentication security.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select Voter login	Voter portal displayed	As Expected	Pass
2	Select Sign In	Login form shown	As Expected	Pass
3	Enter wrong CNIC: 1111111111111	CNIC accepted for input	As Expected	Pass
4	Enter any password: test123	Password accepted for input	As Expected	Pass
5	Submit	Error: Invalid CNIC or password	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
6	Retry with correct CNIC but wrong password	Error message shown	As Expected	Pass
7	Login with correct credentials	Login successful	As Expected	Pass

---

Test Case ID: SYS_022

Test Case Description: Test Empty Field Validation in Registration

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: December 3, 2025
Test Case Status: Fail

Prerequisites:

1. Application running
2. Voter registration page accessed

Test Data:

- Empty fields for name, province, district

Test Scenario:

Verify system validates required fields in registration.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to voter registration	Form displayed	As Expected	Pass
2	Enter valid CNIC only	CNIC accepted	As Expected	Pass
3	Leave name empty, press enter	Error: Name required	No validation, accepts empty	**Fail**
4	Leave province empty	Error: Province required	No validation	**Fail**
5	Submit form with empty fields	Registration should fail	Registration succeeds with empty fields	**Fail**

Bug Report: No validation for empty fields in registration. Can create invalid voter records.

Test Case ID: SYS_023

Test Case Description: Test Empty Field Validation in Registration (Fixed)

Created By: Umair
Reviewed By: Usman
Version: 2.0
Date Tested: December 6, 2025
Test Case Status: Pass

Prerequisites:

1. Validation code added to registerVoter() function
2. Application recompiled

Test Data:

- Valid CNIC: 3520298765432
- Empty name, province, district

Test Scenario:

Verify registration now validates all required fields.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to registration	Form displayed	As Expected	Pass
2	Enter CNIC, leave name empty	Validation error shown	As Expected	Pass
3	Enter name, leave province empty	Validation error shown	As Expected	Pass
4	Enter province, leave district empty	Validation error shown	As Expected	Pass
5	Fill all required fields	Registration proceeds	As Expected	Pass
6	Submit complete form	Success message displayed	As Expected	Pass

Test Case ID: SYS_024

Test Case Description: Test Age Validation for Candidates

Created By: Umair
Reviewed By: Abdul Muiz

Version: 1.0
Date Tested: December 8, 2025
Test Case Status: Fail

Prerequisites:

1. Admin logged in
2. Add candidate form accessed

Test Data:

- Name: Young Candidate
- Age: 17 (below voting age)
- Age: 25 (valid)

Test Scenario:

Verify system validates candidate age requirements.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin selects "Add Candidate"	Form displayed	As Expected	Pass
2	Enter candidate details	Details accepted	As Expected	Pass
3	Enter age: 17	Error: Minimum age 18	No validation, accepts 17	**Fail**
4	Submit form	Should reject	Candidate added successfully	**Fail**

Bug Report: No age validation for candidates. Underage candidates can be added.

Test Case ID: SYS_025

Test Case Description: Test Age Validation for Candidates (Fixed)

Created By: Umair
Reviewed By: Usman
Version: 2.0
Date Tested: December 10, 2025
Test Case Status: Pass

Prerequisites:

1. Age validation added to addCandidate() function

2. Admin logged in

Test Data:

- Age test values: 17, 18, 25, 150

Test Scenario:

Verify candidate age validation now works correctly.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin adds candidate with age 17	Error: Must be 18+	As Expected	Pass
2	Admin adds candidate with age 150	Error: Invalid age	As Expected	Pass
3	Admin adds candidate with age 25	Candidate added successfully	As Expected	Pass
4	Verify database record	Age = 25 stored correctly	As Expected	Pass

---

Test Case ID: SYS_026

Test Case Description: Test Election Status Persistence

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: December 12, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Elections initially stopped

Test Data:

- Province: Punjab

Test Scenario:

Verify election status persists across application sessions.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin starts National election	Status changed to **'start'**	As Expected	Pass
2	Admin starts Local election (Punjab)	Status changed to **'start'**	As Expected	Pass
3	Close application	Application closed	As Expected	Pass
4	Restart application	Application running	As Expected	Pass
5	Login as voter	Voter menu displayed	As Expected	Pass
6	Check voting options	MNA and MPA options available	As Expected	Pass
7	Verify election_status table	Shows **'start'** for both elections	As Expected	Pass

---

Test Case ID: SYS_027

Test Case Description: Test Vote Count Persistence

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: December 15, 2025
Test Case Status: Pass

Prerequisites:

1. Elections active
2. Multiple voters ready to vote
3. Candidate ID 5 with initial votes: 0

Test Data:

- Number of voters: 5
- All voting for Candidate ID: 5

Test Scenario:

Verify vote counts persist correctly in database.

Step #	Step Details	Expected Results	Actual Results	Status
1	5 voters vote for Candidate 5	Each vote recorded	As Expected	Pass
2	Check candidate vote count	Shows 5 votes	As Expected	Pass
3	Restart application	Application restarted	As Expected	Pass
4	Admin views results	Vote count still shows 5	As Expected	Pass
5	Candidate logs in and checks votes	Shows 5 votes	As Expected	Pass

---

Test Case ID: SYS_028

Test Case Description: Test Multiple Province Elections Simultaneously

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: December 18, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Candidates in multiple provinces

Test Data:

- Provinces: Punjab, Sindh, KPK

Test Scenario:

Verify system can handle multiple local elections in different provinces.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin starts Local election for Punjab	Punjab election active	As Expected	Pass
2	Admin starts Local election for Sindh	Sindh election active	As Expected	Pass
3	Admin starts Local election for KPK	KPK election active	As Expected	Pass
4	Punjab voter votes for MPA	Vote recorded	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
5	Sindh voter votes for MPA	Vote recorded	As Expected	Pass
6	KPK voter votes for MPA	Vote recorded	As Expected	Pass
7	Admin views Punjab results	Only Punjab results shown	As Expected	Pass
8	Admin views Sindh results	Only Sindh results shown	As Expected	Pass

Test Case ID: SYS_029

Test Case Description: Test SQL Injection Prevention

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: December 20, 2025
Test Case Status: Fail

Prerequisites:

1. Application running
2. Login screen accessible

Test Data:

- Malicious inputs:
  - CNIC: 3520212345678' OR '1'='1
  - Password: ' OR '1'='1

Test Scenario:

Verify system prevents SQL injection attacks.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to voter login	Login form displayed	As Expected	Pass
2	Enter CNIC with SQL injection: 3520212345678' OR '1'='1	Input sanitized/rejected	Input accepted	**Fail**

Step #	Step Details	Expected Results	Actual Results	Status
3	Enter password: ' OR '1'='1	Input sanitized/rejected	Input accepted	**Fail**
4	Submit form	Login should fail	Potential security breach	**Fail**

Bug Report: SQL injection vulnerability. Input not sanitized. Critical security issue.

Test Case ID: SYS_030

Test Case Description: Test SQL Injection Prevention (Fixed)

Created By: Umair
Reviewed By: Usman
Version: 2.0
Date Tested: December 23, 2025
Test Case Status: Pass

Prerequisites:

1. Prepared statements implemented
2. Input sanitization added
3. Application recompiled

Test Data:

- Malicious CNIC: 3520212345678' OR '1'='1
- Malicious Password: ' OR '1'='1
- Valid credentials for comparison

Test Scenario:

Verify SQL injection protection after implementing prepared statements.

Step #	Step Details	Expected Results	Actual Results	Status
1	Attempt login with SQL injection in CNIC	Login fails securely	As Expected	Pass
2	Attempt SQL injection in password field	Login fails securely	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
3	Check database logs	No unauthorized access	As Expected	Pass
4	Login with valid credentials	Login successful	As Expected	Pass
5	Attempt injection in candidate addition	Input sanitized	As Expected	Pass

---

Test Case ID: SYS_031

Test Case Description: Test Password Display Security

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: December 26, 2025
Test Case Status: Fail

Prerequisites:

1. Application running
2. Login screens accessible

Test Data:

- Password: TestPass123

Test Scenario:

Verify passwords are masked during input.

Step #	Step Details	Expected Results	Actual Results	Status
1	Navigate to any login screen	Login form shown	As Expected	Pass
2	Enter password	Password masked with * or •	Password visible in plaintext	**Fail**
3	Observer can see password	Should not be visible	Visible to observer	**Fail**

Bug Report: Password input not masked. Security vulnerability - shoulder surfing risk.

Test Case ID: SYS_032

Test Case Description: Test Logout Functionality

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: December 28, 2025
Test Case Status: Pass

Prerequisites:

1. User logged in (any type: admin, voter, candidate)

Test Data:

- Various user types

Test Scenario:

Verify logout properly terminates session and returns to main menu.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as Admin	Admin menu displayed	As Expected	Pass
2	Select Logout	Returned to main menu	As Expected	Pass
3	Login as Voter	Voter menu displayed	As Expected	Pass
4	Select Logout	Returned to main menu	As Expected	Pass
5	Login as Candidate	Candidate menu displayed	As Expected	Pass
6	Select Logout	Returned to main menu	As Expected	Pass
7	Verify session cleared	Cannot access previous menu without login	As Expected	Pass

---

Test Case ID: SYS_033

Test Case Description: Test Results Accuracy After Multiple Votes

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: January 2, 2026
Test Case Status: Pass

Prerequisites:

1. National election active
2. 10 registered voters
3. 5 MNA candidates

Test Data:

- Candidate 1: expects 4 votes
- Candidate 2: expects 3 votes
- Candidate 3: expects 2 votes
- Candidate 4: expects 1 vote
- Candidate 5: expects 0 votes

Test Scenario:

Verify vote counting accuracy with multiple voters.

Step #	Step Details	Expected Results	Actual Results	Status
1	10 voters cast votes as per data	All votes recorded	As Expected	Pass
2	Admin views National results	Results page displayed	As Expected	Pass
3	Verify Candidate 1 count	Shows 4 votes	As Expected	Pass
4	Verify Candidate 2 count	Shows 3 votes	As Expected	Pass
5	Verify Candidate 3 count	Shows 2 votes	As Expected	Pass
6	Verify Candidate 4 count	Shows 1 vote	As Expected	Pass
7	Verify Candidate 5 count	Shows 0 votes	As Expected	Pass
8	Check results sorting	Sorted by votes descending	As Expected	Pass

---

Test Case ID: SYS_034

Test Case Description: Test System Performance with Large Dataset

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: January 5, 2026
Test Case Status: Pass

Prerequisites:

1. Database populated with:
    - 1000 voters
    - 100 candidates
    - Multiple completed votes

Test Data:

- Large dataset as above

Test Scenario:

Verify system maintains acceptable performance with large data volumes.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as voter	Response time < 3 seconds	1.2 seconds	Pass
2	View results	Response time < 5 seconds	2.8 seconds	Pass
3	Admin views all candidates	Loads within 5 seconds	3.5 seconds	Pass
4	Cast vote	Response time < 2 seconds	0.9 seconds	Pass
5	Calculate winners	Completes within 10 seconds	6.2 seconds	Pass

---

Test Case ID: SYS_035

Test Case Description: Test Data Integrity After Power Failure Simulation

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: January 8, 2026
Test Case Status: Pass

Prerequisites:

1. Election active with votes being cast
2. Ability to stop MySQL service suddenly

Test Data:

- Candidate 10: initial votes = 5
- 3 votes cast successfully
- Expected final: 8 votes

Test Scenario:

Verify database transaction integrity.

Step #	Step Details	Expected Results	Actual Results	Status
1	Note initial vote count for Candidate 10	5 votes	As Expected	Pass
2	Voter 1 votes for Candidate 10	Vote recorded, count = 6	As Expected	Pass
3	Voter 2 votes for Candidate 10	Vote recorded, count = 7	As Expected	Pass
4	During Voter 3's vote, simulate crash	Transaction in progress	As Expected	Pass
5	Restart MySQL and application	System recovers	As Expected	Pass
6	Check Candidate 10 vote count	Either 7 or 8 (transaction completed or rolled back)	7 (rolled back)	Pass
7	Verify voter 3 status	has_voted should match vote count	Consistent	Pass

---

Test Case ID: SYS_036

Test Case Description: Test Cross-Province Voting Prevention

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: January 10, 2026
Test Case Status: Pass

Prerequisites:

1. Voter registered in Punjab
2. Candidates exist in Sindh
3. National election active

Test Data:

- Voter Province: Punjab
- Attempt to vote for: Sindh candidate

Test Scenario:

Verify voters cannot vote for candidates outside their region.

Step #	Step Details	Expected Results	Actual Results	Status
1	Login as Punjab voter	Login successful	As Expected	Pass
2	Select "Vote for MNA"	Only Punjab candidates shown	As Expected	Pass
3	Manually attempt Sindh candidate ID	Vote rejected or not possible	Not possible (ID not in list)	Pass
4	Local election: try Multan candidate	Not shown (voter is Lahore)	As Expected	Pass

---

Test Case ID: SYS_037

Test Case Description: Test Election Results Before Voting Ends

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: January 12, 2026
Test Case Status: Pass

Prerequisites:

1. Elections active
2. Some votes cast but election not ended

Test Data:

- National election active
- 10 votes cast out of 100 voters

Test Scenario:

Verify results can be viewed during active election (transparency).

Step #	Step Details	Expected Results	Actual Results	Status
1	Election in progress	Status = 'start'	As Expected	Pass
2	Voter views results	Current standings shown	As Expected	Pass
3	Admin views results	Current vote counts visible	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
4	Candidate checks own votes	Current vote count shown	As Expected	Pass
5	Cast more votes	Results update accordingly	As Expected	Pass

---

Test Case ID: SYS_038

Test Case Description: Test Boundary Values for Numeric Inputs

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: January 15, 2026
Test Case Status: Pass

Prerequisites:

1. Various forms accessible
2. Validator class functioning

Test Data:

- Age: 0, 18, 150, 999
- Menu choices: 0, 1, 99, -1
- Candidate IDs: 0, 1, 999999

Test Scenario:

Verify boundary value handling in numeric inputs.

Step #	Step Details	Expected Results	Actual Results	Status
1	Enter age 0 in candidate form	Rejected as invalid	As Expected	Pass
2	Enter age 18	Accepted (minimum valid)	As Expected	Pass
3	Enter age 150	Rejected as unrealistic	As Expected	Pass
4	Menu input: 0	Invalid choice message	As Expected	Pass
5	Menu input: 1	Valid, proceeds	As Expected	Pass
6	Menu input: 999	Invalid choice message	As Expected	Pass

Test Case ID: SYS_039

Test Case Description: Test Special Characters in Text Fields

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: January 18, 2026
Test Case Status: Fail

Prerequisites:

1. Registration or candidate addition form

Test Data:

- Names with special characters: O'Brien, José, Müller
- Province: Punjab@123

Test Scenario:

Verify handling of special characters in text inputs.

Step #	Step Details	Expected Results	Actual Results	Status
1	Enter name: O'Brien	Accepted or sanitized	Causes SQL error	**Fail**
2	Enter name: José	Accepted (valid Unicode)	Not properly stored	**Fail**
3	Enter province: Punjab@123	Rejected (invalid format)	Accepted	**Fail**

Bug Report: Special characters not handled properly. SQL errors and data corruption possible.

Test Case ID: SYS_040

Test Case Description: Test Menu Navigation Flow

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: January 20, 2026
Test Case Status: Pass

Prerequisites:

1. Application running
2. All user types available

Test Data:

- Multiple navigation paths

Test Scenario:

Verify user can navigate through all menu options and return appropriately.

Step #	Step Details	Expected Results	Actual Results	Status
1	From main menu, select Admin	Admin login shown	As Expected	Pass
2	Login as admin	Admin menu shown	As Expected	Pass
3	Access each admin menu option	Each function accessible	As Expected	Pass
4	Logout	Return to main menu	As Expected	Pass
5	Select Voter	Voter portal shown	As Expected	Pass
6	Sign up new voter	Registration successful	As Expected	Pass
7	Logout	Return to main menu	As Expected	Pass
8	Select Candidate	Candidate type selection shown	As Expected	Pass
9	Cancel/back	Return to main menu	As Expected	Pass
10	Select Exit	Application terminates cleanly	As Expected	Pass

---

Test Case ID: SYS_041

Test Case Description: Test Voter Status Tracking

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: January 22, 2026
Test Case Status: Pass

Prerequisites:

1. Voter registered and logged in
2. Both election types active

Test Data:

- Voter CNIC: 3520212345678

Test Scenario:

Verify system accurately tracks which elections voter has participated in.

Step #	Step Details	Expected Results	Actual Results	Status
1	Check initial status	has_voted_mna=0, has_voted_mpa=0	As Expected	Pass
2	Vote for MNA	Vote cast successfully	As Expected	Pass
3	Check database	has_voted_mna=1, has_voted_mpa=0	As Expected	Pass
4	Logout and login again	Previous voting status maintained	As Expected	Pass
5	MNA option no longer available	Menu shows option disabled	As Expected	Pass
6	Vote for MPA	Vote cast successfully	As Expected	Pass
7	Check database	has_voted_mna=1, has_voted_mpa=1	As Expected	Pass
8	Both voting options disabled	No voting options available	As Expected	Pass

---

Test Case ID: SYS_042

Test Case Description: Test Database Connection Pool Under Load

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: January 25, 2026
Test Case Status: Pass

Prerequisites:

1. Multiple application instances
2. Load testing capability

Test Data:

- 20 simultaneous users

Test Scenario:

Verify database connections are managed properly under concurrent load.

Step #	Step Details	Expected Results	Actual Results	Status
1	Launch 20 application instances	All launch successfully	As Expected	Pass
2	All instances login simultaneously	All login successfully	As Expected	Pass
3	All query database at same time	All queries complete	As Expected	Pass
4	Check MySQL connection count	Within acceptable limits	As Expected	Pass
5	All instances perform voting	All votes recorded	As Expected	Pass
6	Check for connection errors	No errors logged	As Expected	Pass

---

Test Case ID: SYS_043

Test Case Description: Test Election Closure Impact on Active Voters

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: January 28, 2026
Test Case Status: Pass

Prerequisites:

1. National election active
2. Voter logged in and viewing candidates

Test Data:

- Active voter session

Test Scenario:

Verify behavior when election is closed while voter is in voting process.

Step #	Step Details	Expected Results	Actual Results	Status
1	Voter logs in, election active	Voter menu with voting option shown	As Expected	Pass
2	Voter selects "Vote for MNA"	Candidate list displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
3	Admin (separate session) stops election	Election status changed	As Expected	Pass
4	Voter attempts to cast vote	Should be prevented or warned	Vote prevented with message	Pass
5	Voter returns to menu	Voting option no longer available	As Expected	Pass

Test Case ID: SYS_044

Test Case Description: Test Results Display Formatting

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: January 30, 2026
Test Case Status: Pass

Prerequisites:

1. Multiple candidates with votes
2. Elections completed

Test Data:

- 10 candidates with varying vote counts
- Long candidate names
- Long party names

Test Scenario:

Verify results are displayed in readable, well-formatted manner.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin views results	Results page displayed	As Expected	Pass
2	Check column alignment	All columns aligned properly	As Expected	Pass
3	Check long names	No text overflow or cutoff	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
4	Verify sorting	Candidates sorted by votes descending	As Expected	Pass
5	Check readability	Clear separation between rows	As Expected	Pass
6	Voter views results	Same formatting quality	As Expected	Pass

---

Test Case ID: SYS_045

Test Case Description: Test Help/Guidance for New Users

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: February 2, 2026
Test Case Status: Fail

Prerequisites:

1. Fresh user accessing system first time

Test Data:

- New user perspective

Test Scenario:

Verify system provides adequate guidance for first-time users.

Step #	Step Details	Expected Results	Actual Results	Status
1	Launch application	Main menu shown	As Expected	Pass
2	Look for help option	Help/Instructions available	No help option found	**Fail**
3	Try invalid input	Clear error message with guidance	Error shown but no guidance	**Fail**
4	Registration process	Step-by-step instructions	Minimal guidance	**Fail**

Bug Report: Poor user experience for new users. No help system or detailed instructions.

Test Case ID: SYS_046

Test Case Description: Test Tie-Breaking in Elections

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: February 5, 2026
Test Case Status: Pass

Prerequisites:

1. National election with provinces
2. Equal provincial wins scenario possible

Test Data:

- Punjab: PartyA wins
- Sindh: PartyB wins
- Tie scenario: 1-1

Test Scenario:

Verify system handles tie scenarios in PM selection.

Step #	Step Details	Expected Results	Actual Results	Status
1	Set up tie scenario (2 provinces, each won by different party)	Data configured	As Expected	Pass
2	Complete elections	All votes recorded	As Expected	Pass
3	Admin views National results	Results calculated	As Expected	Pass
4	Check PM announcement	"Tie detected. Manual intervention required."	As Expected	Pass
5	Verify no automatic winner	No PM declared	As Expected	Pass

Test Case ID: SYS_047

Test Case Description: Test Application Exit and Resource Cleanup

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: February 8, 2026
Test Case Status: Pass

Prerequisites:

1. Application running
2. Database connections active

Test Data:

- None

Test Scenario:

Verify proper resource cleanup on application exit.

Step #	Step Details	Expected Results	Actual Results	Status
1	Application running normally	All features working	As Expected	Pass
2	Select Exit from main menu	Cleanup message shown	As Expected	Pass
3	Check database disconnect message	"Database disconnected." shown	As Expected	Pass
4	Check MySQL connections	Connection closed	As Expected	Pass
5	Application terminates	Clean exit, no hanging processes	As Expected	Pass
6	Check for memory leaks (if tools available)	No leaks detected	As Expected	Pass

---

Test Case ID: SYS_048

Test Case Description: Test Party Affiliation Consistency

Created By: Umair
Reviewed By: Usman
Version: 1.0
Date Tested: February 10, 2026
Test Case Status: Pass

Prerequisites:

1. Multiple candidates from same party
2. Elections active

Test Data:

- Party: PartyA
- 5 candidates from PartyA across different regions

Test Scenario:

Verify party affiliations are tracked correctly across all regions.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin adds candidates from PartyA	All added successfully	As Expected	Pass
2	Verify party names in database	All show "PartyA" consistently	As Expected	Pass
3	Conduct elections	Votes recorded	As Expected	Pass
4	View results by province	PartyA candidates grouped correctly	As Expected	Pass
5	Calculate PM winner	Party wins counted accurately	As Expected	Pass

---

Test Case ID: SYS_049

Test Case Description: Test Error Recovery Mechanisms

Created By: Umair
Reviewed By: Abdul Muiz
Version: 1.0
Date Tested: February 12, 2026
Test Case Status: Pass

Prerequisites:

1. Various error conditions can be simulated

Test Data:

- Database query failures
- Invalid data scenarios

Test Scenario:

Verify application recovers gracefully from various errors.

Step #	Step Details	Expected Results	Actual Results	Status
1	Simulate database query failure	Error caught and handled	As Expected	Pass
2	Check user notification	Clear error message shown	As Expected	Pass
3	Verify application continues	Returns to menu, still functional	As Expected	Pass
4	Simulate invalid candidate ID	Error handled gracefully	As Expected	Pass
5	Test invalid province name	Appropriate error message	As Expected	Pass
6	Return to normal operation	System recovers fully	As Expected	Pass

---

Test Case ID: SYS_050

Test Case Description: Test Complete System Integration End-to-End

Created By: Umair
Reviewed By: Usman & Abdul Muiz
Version: 1.0
Date Tested: February 15, 2026
Test Case Status: Pass

Prerequisites:

1. Fresh database with minimal data
2. Full system functional

Test Data:

- 3 provinces: Punjab, Sindh, KPK
- 15 candidates (5 MNA, 10 MPA across provinces)
- 20 voters from different regions

Test Scenario:

Comprehensive end-to-end test of entire voting system lifecycle.

Step #	Step Details	Expected Results	Actual Results	Status
1	Admin adds all candidates	All 15 candidates added	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Status
2	20 voters register	All registrations successful	As Expected	Pass
3	Admin starts National election	Election active	As Expected	Pass
4	Admin starts Local elections (all provinces)	All active	As Expected	Pass
5	All voters cast MNA votes	All votes recorded	As Expected	Pass
6	All voters cast MPA votes	All votes recorded	As Expected	Pass
7	Candidates check their votes	All can view their counts	As Expected	Pass
8	Admin stops all elections	All elections stopped	As Expected	Pass
9	Admin views National results	PM declared based on provincial wins	As Expected	Pass
10	Admin views Local results for each province	CMs declared for each	As Expected	Pass
11	Verify all vote counts accurate	Database matches expected	As Expected	Pass
12	Verify voter statuses correct	All marked as voted	As Expected	Pass
13	System performs efficiently	No delays or crashes	As Expected	Pass
14	Data integrity maintained	All data consistent	As Expected	Pass

UNIT LEVEL TESTING

Tester: Abdul Muiz

Test Case ID: UNIT_001

Test Case Description: Test Validator.getIntInput() with valid integer
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 2, 2025
Test Case Status: Fail

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input string: "123"
- Expected integer: 123

Test Scenario: Verify getIntInput() converts valid numeric string to integer correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call getIntInput() with "123"	Returns true	Returns false (logic error)	Fail
2	Check output integer value	val = 123	val = undefined	Fail
3	Verify no error message	No output	Error message shown	Fail

Test Case ID: UNIT_002

Test Case Description: Test Validator.getIntInput() with valid integer (Fixed)
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: October 5, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled with bug fixes
2. Test harness setup

Test Data:

- Input string: "123"
- Expected integer: 123

Test Scenario: Verify getIntInput() converts valid numeric string to integer correctly after bug fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call getIntInput() with "123"	Returns true	As Expected	Pass
2	Check output integer value	val = 123	As Expected	Pass
3	Verify no error message	No output	As Expected	Pass

Test Case ID: UNIT_003

Test Case Description: Test Validator.getIntInput() with invalid string
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: October 5, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input string: "abc123"
- Expected result: false

Test Scenario: Verify getIntInput() rejects non-numeric strings.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call getIntInput() with "abc123"	Returns false	As Expected	Pass
2	Check error message displayed	"Invalid input. Only numbers are allowed."	As Expected	Pass
3	Verify value not modified	val unchanged	As Expected	Pass

Test Case ID: UNIT_004

Test Case Description: Test Validator.isAlpha() with alphabetic string
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 8, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input string: "Muhammad"

Test Scenario: Verify isAlpha() returns true for alphabetic strings.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call isAlpha("Muhammad")	Returns true	As Expected	Pass
2	Call isAlpha("PAKISTAN")	Returns true	As Expected	Pass
3	Verify mixed case handling	Returns true	As Expected	Pass

Test Case ID: UNIT_005

Test Case Description: Test Validator.isAlpha() with non-alphabetic string
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 8, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input strings: "Ali123", "Test@Name", "123"

Test Scenario: Verify isAlpha() returns false for strings with numbers or special characters.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call isAlpha("Ali123")	Returns false	As Expected	Pass
2	Call isAlpha("Test@Name")	Returns false	As Expected	Pass
3	Call isAlpha("123")	Returns false	As Expected	Pass

Test Case ID: UNIT_006

Test Case Description: Test Validator.isNumeric() with numeric string
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0

Date Tested: October 10, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input string: "3520212345678"

Test Scenario: Verify isNumeric() validates CNIC format correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call isNumeric("3520212345678")	Returns true	As Expected	Pass
2	Call isNumeric("1234567890")	Returns true	As Expected	Pass
3	Verify digit-only validation	Returns true	As Expected	Pass

Test Case ID: UNIT_007

Test Case Description: Test Validator.isNumeric() with non-numeric string
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 10, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input strings: "35202-1234567-8", "ABC123", "12.34"

Test Scenario: Verify isNumeric() rejects strings with non-digits.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call isNumeric("35202-1234567-8")	Returns false	As Expected	Pass
2	Call isNumeric("ABC123")	Returns false	As Expected	Pass
3	Call isNumeric("12.34")	Returns false	As Expected	Pass

Test Case ID: UNIT_008

Test Case Description: Test Validator.isAlphanumeric() with valid input
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 12, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input strings: "User123", "Admin001", "Voter99"

Test Scenario: Verify isAlphanumeric() accepts letters and numbers.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call isAlphanumeric("User123")	Returns true	As Expected	Pass
2	Call isAlphanumeric("Admin001")	Returns true	As Expected	Pass
3	Call isAlphanumeric("Voter99")	Returns true	As Expected	Pass

Test Case ID: UNIT_009

Test Case Description: Test Validator.isAlphanumeric() with special characters
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 12, 2025
Test Case Status: Pass

Prerequisites:

1. Validator class compiled
2. Test harness setup

Test Data:

- Input strings: "User@123", "Admin-001", "Voter_99"

Test Scenario: Verify isAlphanumeric() rejects strings with special characters.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Call isAlphanumeric("User@123")	Returns false	As Expected	Pass
2	Call isAlphanumeric("Admin-001")	Returns false	As Expected	Pass
3	Call isAlphanumeric("Voter_99")	Returns false	As Expected	Pass

---

Test Case ID: UNIT_010

Test Case Description: Test DatabaseManager.connect() functionality
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 15, 2025
Test Case Status: Fail

Prerequisites:

1. MySQL server not running
2. DatabaseManager class compiled

Test Data:

- Host: localhost
- User: root
- Password: ali@123
- Database: votingsystem

Test Scenario: Verify DatabaseManager handles connection failure gracefully.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Stop MySQL server	Server stopped	As Expected	Pass
2	Call DatabaseManager::connect()	Error message displayed	Program crashes	Fail
3	Verify error handling	Graceful failure	Exit(1) called	Fail

Test Case ID: UNIT_011

Test Case Description: Test DatabaseManager.connect() with proper error handling
Created By: Abdul Muiz

Reviewed By: Usman
Version: 1.1
Date Tested: October 18, 2025
Test Case Status: Pass

Prerequisites:

1. MySQL server running
2. DatabaseManager class compiled with improved error handling

Test Data:

- Host: localhost
- User: root
- Password: ali@123
- Database: votingsystem

Test Scenario: Verify DatabaseManager connects successfully with proper error handling.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Start MySQL server	Server running	As Expected	Pass
2	Call DatabaseManager::connect()	"Database connection successful"	As Expected	Pass
3	Verify connection object	conn != nullptr	As Expected	Pass

Test Case ID: UNIT_012

Test Case Description: Test DatabaseManager.disconnect() functionality
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: October 18, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. DatabaseManager class compiled

Test Data:

- Active connection established

Test Scenario: Verify DatabaseManager disconnects properly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Establish connection	Connection successful	As Expected	Pass
2	Call DatabaseManager::disconnect()	"Database disconnected"	As Expected	Pass
3	Verify connection closed	conn closed properly	As Expected	Pass

---

Test Case ID: UNIT_013

Test Case Description: Test Admin.login() with valid credentials
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 20, 2025
Test Case Status: Pass

Prerequisites:

1. Admin class instantiated
2. Input stream available

Test Data:

- Username: admin
- Password: admin123

Test Scenario: Verify Admin can login with correct credentials.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter username "admin"	Username accepted	As Expected	Pass
2	Enter password "admin123"	Password accepted	As Expected	Pass
3	Call login()	Returns true, "Login successful!"	As Expected	Pass

---

Test Case ID: UNIT_014

Test Case Description: Test Admin.login() with invalid credentials
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 20, 2025
Test Case Status: Pass

Prerequisites:

1. Admin class instantiated
2. Input stream available

Test Data:

- Username: wronguser
- Password: wrongpass

Test Scenario: Verify Admin login fails with incorrect credentials.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter username "wronguser"	Username accepted	As Expected	Pass
2	Enter password "wrongpass"	Password accepted	As Expected	Pass
3	Call login()	Returns false, "Invalid admin credentials"	As Expected	Pass

---

Test Case ID: UNIT_015

Test Case Description: Test LocalElection.start() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 22, 2025
Test Case Status: Fail

Prerequisites:

1. Database connected
2. election_status table exists
3. LocalElection object created

Test Data:

- Province: "Punjab"

Test Scenario: Verify LocalElection starts correctly for a province.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create LocalElection("Punjab")	Object created	As Expected	Pass
2	Call start()	Election status updated	SQL syntax error	Fail
3	Verify database update	status='start', level='local'	Query failed	Fail

---

Test Case ID: UNIT_016

Test Case Description: Test LocalElection.start() method (Fixed)
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: October 25, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. election_status table schema corrected
3. LocalElection object created

Test Data:

- Province: "Punjab"

Test Scenario: Verify LocalElection starts correctly after bug fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create LocalElection("Punjab")	Object created	As Expected	Pass
2	Call start()	"Local election started for Punjab"	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Verify database update	status='start', level='local', province='Punjab'	As Expected	Pass

Test Case ID: UNIT_017

Test Case Description: Test LocalElection.end() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: October 25, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. Local election started for "Sindh"
3. LocalElection object created

Test Data:

- Province: "Sindh"

Test Scenario: Verify LocalElection ends correctly for a province.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create LocalElection("Sindh")	Object created	As Expected	Pass
2	Call end()	"Local election ended for Sindh"	As Expected	Pass
3	Verify database update	status='stop', level='district'	As Expected	Pass

Test Case ID: UNIT_018

Test Case Description: Test NationalElection.start() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0

Date Tested: October 28, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. election_status table exists
3. NationalElection object created

Test Data:

- None required

Test Scenario: Verify NationalElection starts correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create NationalElection object	Object created	As Expected	Pass
2	Call start()	"National election started"	As Expected	Pass
3	Verify database update	status='start', level='national', province=''	As Expected	Pass

---

Test Case ID: UNIT_019

Test Case Description: Test NationalElection.end() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 28, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. National election started
3. NationalElection object created

Test Data:

- None required

Test Scenario: Verify NationalElection ends correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create NationalElection object	Object created	As Expected	Pass
2	Call end()	"National election ended"	As Expected	Pass
3	Verify database update	status='stop', level='provincial'	As Expected	Pass

---

Test Case ID: UNIT_020

Test Case Description: Test NationalElection.countWinner() with PartyA majority
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: October 30, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MNA candidates with votes populated
3. PartyA has majority

Test Data:

- PartyA wins: 3 provinces
- PartyB wins: 1 province

Test Scenario: Verify countWinner() correctly identifies PartyA as winner.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Populate test data	PartyA majority set	As Expected	Pass
2	Call countWinner()	"Prime Minister from PartyA"	As Expected	Pass
3	Verify return value	Returns 0	As Expected	Pass

Test Case ID: UNIT_021

Test Case Description: Test NationalElection.countWinner() with tie scenario
Created By: Abdul Muiz
Reviewed By: Usman

Version: 1.0
Date Tested: October 30, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MNA candidates with votes populated
3. Equal wins for both parties

Test Data:

- PartyA wins: 2 provinces
- PartyB wins: 2 provinces

Test Scenario: Verify countWinner() handles tie correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Populate tie data	Equal distribution set	As Expected	Pass
2	Call countWinner()	"Tie detected. Manual intervention required"	As Expected	Pass
3	Verify return value	Returns -1	As Expected	Pass

Test Case ID: UNIT_022

Test Case Description: Test MNA.viewDetails() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 2, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MNA candidate logged in with ID = 5
3. Candidate data exists

Test Data:

- Candidate ID: 5

- Name: "Ahmed Khan"
- Party: "PartyA"

Test Scenario: Verify MNA can view their details correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as MNA with ID 5	Login successful	As Expected	Pass
2	Call viewDetails()	Display name, party, province, district, age	As Expected	Pass
3	Verify data accuracy	Data matches database	As Expected	Pass

---

Test Case ID: UNIT_023

Test Case Description: Test MNA.viewVotes() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 2, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MNA candidate logged in with ID = 5
3. Vote count exists

Test Data:

- Candidate ID: 5
- Votes: 1250

Test Scenario: Verify MNA can view their vote count.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as MNA with ID 5	Login successful	As Expected	Pass
2	Call viewVotes()	"Your current vote count: 1250"	As Expected	Pass
3	Verify count accuracy	Count matches database	As Expected	Pass

---

Test Case ID: UNIT_024

Test Case Description: Test MPA.viewDetails() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 5, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MPA candidate logged in with ID = 12
3. Candidate data exists

Test Data:

- Candidate ID: 12
- Name: "Fatima Ali"
- Party: "PartyB"

Test Scenario: Verify MPA can view their details correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as MPA with ID 12	Login successful	As Expected	Pass
2	Call viewDetails()	Display name, party, province, district, age	As Expected	Pass
3	Verify data accuracy	Data matches database	As Expected	Pass

---

Test Case ID: UNIT_025

Test Case Description: Test MPA.viewVotes() method
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 5, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected

2. MPA candidate logged in with ID = 12
3. Vote count exists

Test Data:

- Candidate ID: 12
- Votes: 850

Test Scenario: Verify MPA can view their vote count.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as MPA with ID 12	Login successful	As Expected	Pass
2	Call viewVotes()	"Your current vote count: 850"	As Expected	Pass
3	Verify count accuracy	Count matches database	As Expected	Pass

---

Test Case ID: UNIT_026

Test Case Description: Test Candidate.login() with invalid ID
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 8, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MNA object created

Test Data:

- Candidate ID: 9999
- Password: "anypassword"

Test Scenario: Verify Candidate login fails with non-existent ID.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter ID 9999	ID accepted	As Expected	Pass
2	Enter any password	Password accepted	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Call login()	Returns false, "Invalid ID or password"	As Expected	Pass

---

Test Case ID: UNIT_027

Test Case Description: Test Candidate.login() with wrong password
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 8, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MNA object created
3. Candidate ID 5 exists

Test Data:

- Candidate ID: 5
- Password: "wrongpassword"
- Correct password: "pass123"

Test Scenario: Verify Candidate login fails with incorrect password.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter valid ID 5	ID accepted	As Expected	Pass
2	Enter wrong password	Password accepted	As Expected	Pass
3	Call login()	Returns false, "Invalid ID or password"	As Expected	Pass

---

Test Case ID: UNIT_028

Test Case Description: Test Voter CNIC length validation
Created By: Abdul Muiz

Reviewed By: Usman
Version: 1.0
Date Tested: November 10, 2025
Test Case Status: Fail

Prerequisites:

1. Voter registration initiated
2. Database connected

Test Data:

- CNIC: "12345" (5 digits)

Test Scenario: Verify registration rejects CNIC with incorrect length.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter CNIC "12345"	CNIC accepted for validation	As Expected	Pass
2	Validate length	Validation should fail	No validation performed	Fail
3	Display error message	"Invalid CNIC format. Must be 13 digits"	No message shown	Fail

---

Test Case ID: UNIT_029

Test Case Description: Test Voter CNIC length validation (Fixed)
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: November 12, 2025
Test Case Status: Pass

Prerequisites:

1. Voter registration with improved validation
2. Database connected

Test Data:

- CNIC: "12345" (5 digits)

Test Scenario: Verify registration rejects CNIC with incorrect length after fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter CNIC "12345"	CNIC accepted for validation	As Expected	Pass
2	Validate length	Validation fails	As Expected	Pass
3	Display error message	"Invalid CNIC format. Must be 13 digits"	As Expected	Pass

---

Test Case ID: UNIT_030

Test Case Description: Test Voter registration with valid CNIC
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: November 12, 2025
Test Case Status: Pass

Prerequisites:

1. Voter registration form
2. Database connected
3. CNIC not previously registered

Test Data:

- CNIC: "3520212345678"
- Name: "Ali Raza"
- Province: "Punjab"
- District: "Lahore"
- Password: "voter123"

Test Scenario: Verify voter can register successfully with valid data.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter valid CNIC	CNIC validation passes	As Expected	Pass
2	Enter personal details	All fields accepted	As Expected	Pass
3	Submit registration	"Registration successful!"	As Expected	Pass
4	Verify database entry	Voter record created	As Expected	Pass

Test Case ID: UNIT_031

Test Case Description: Test Voter registration with duplicate CNIC
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: November 15, 2025
Test Case Status: Pass

Prerequisites:

1. Voter registration form
2. Database connected
3. CNIC "3520212345678" already registered

Test Data:

- CNIC: "3520212345678" (duplicate)
- Name: "Hassan Ahmed"

Test Scenario: Verify system prevents duplicate CNIC registration.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter existing CNIC	CNIC accepted for validation	As Expected	Pass
2	Check database for duplicate	Duplicate found	As Expected	Pass
3	Display error message	"This CNIC is already registered"	As Expected	Pass
4	Reject registration	Registration not completed	As Expected	Pass

Test Case ID: UNIT_032

Test Case Description: Test Voter.login() with valid credentials
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 18, 2025
Test Case Status: Pass

Prerequisites:

1. Voter registered in database
2. Voter login interface

Test Data:

- CNIC: "3520212345678"
- Password: "voter123"

Test Scenario: Verify voter can login with correct credentials.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Navigate to voter login	Login form displayed	As Expected	Pass
2	Enter CNIC and password	Credentials accepted	As Expected	Pass
3	Submit login	"Login successful. Welcome, Ali Raza!"	As Expected	Pass
4	Verify session state	isLoggedIn = true	As Expected	Pass

Test Case ID: UNIT_033

Test Case Description: Test Voter.login() with invalid password
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 18, 2025
Test Case Status: Pass

Prerequisites:

1. Voter registered in database
2. Voter login interface

Test Data:

- CNIC: "3520212345678"
- Password: "wrongpass"

Test Scenario: Verify voter login fails with incorrect password.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Navigate to voter login	Login form displayed	As Expected	Pass
2	Enter valid CNIC, wrong password	Credentials accepted	As Expected	Pass
3	Submit login	"Invalid CNIC or password"	As Expected	Pass
4	Verify session state	Is logged in = false	As Expected	Pass

Test Case ID: UNIT_034

Test Case Description: Test Admin.addCandidate() with valid data
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 20, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Database connected

Test Data:

- Name: "Imran Shah"
- Party: "PartyA"
- Type: "MNA"
- Province: "KPK"
- District: "Peshawar"
- Age: 45

Test Scenario: Verify admin can add candidate successfully.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Add Candidate" option	Form displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
2	Enter candidate details	All fields accepted	As Expected	Pass
3	Submit form	"Candidate added successfully"	As Expected	Pass
4	Verify database entry	Candidate record created with votes=0	As Expected	Pass

---

Test Case ID: UNIT_035

Test Case Description: Test Admin.addCandidate() with invalid age
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 20, 2025
Test Case Status: Fail

Prerequisites:

1. Admin logged in
2. Database connected

Test Data:

- Name: "Young Candidate"
- Age: "abc" (invalid input)

Test Scenario: Verify admin cannot add candidate with invalid age.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Add Candidate" option	Form displayed	As Expected	Pass
2	Enter age as "abc"	Validation error	Accepted without validation	Fail
3	Attempt submission	Should be blocked	Database error occurs	Fail

---

Test Case ID: UNIT_036

Test Case Description: Test Admin.addCandidate() with input validation (Fixed)
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: November 22, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in with improved validation
2. Database connected

Test Data:

- Name: "Valid Candidate"
- Age: "abc" (invalid input)

Test Scenario: Verify admin cannot add candidate with invalid age after validation fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Add Candidate" option	Form displayed	As Expected	Pass
2	Enter age as "abc"	Validator called	As Expected	Pass
3	Validation check	"Invalid input. Only numbers are allowed"	As Expected	Pass
4	Submission blocked	Form not submitted	As Expected	Pass

Test Case ID: UNIT_037

Test Case Description: Test Admin.removeCandidate() with valid ID
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 25, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Candidate with ID 15 exists

3. Database connected

Test Data:

- Candidate ID: 15

Test Scenario: Verify admin can remove candidate successfully.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Remove Candidate" option	Prompt for ID displayed	As Expected	Pass
2	Enter ID 15	ID accepted	As Expected	Pass
3	Execute deletion	"Candidate removed successfully"	As Expected	Pass
4	Verify database	Record deleted	As Expected	Pass

Test Case ID: UNIT_038

Test Case Description: Test Admin.removeCandidate() with non-existent ID
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 25, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Database connected

Test Data:

- Candidate ID: 9999 (non-existent)

Test Scenario: Verify system handles removal of non-existent candidate gracefully.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Remove Candidate" option	Prompt for ID displayed	As Expected	Pass
2	Enter ID 9999	ID accepted	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Execute deletion	"Candidate removed successfully" (0 rows affected)	As Expected	Pass
4	Verify database	No changes made	As Expected	Pass

Test Case ID: UNIT_039

Test Case Description: Test Voter.isMNAVotingOpen() when voting is closed
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 28, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. National election status = 'stop'
3. Voter logged in

Test Data:

- Election status: stop

Test Scenario: Verify method correctly identifies MNA voting is closed.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Set election status to 'stop'	Status updated in DB	As Expected	Pass
2	Call isMNAVotingOpen()	Returns false	As Expected	Pass
3	Verify query execution	No errors	As Expected	Pass

Test Case ID: UNIT_040

Test Case Description: Test Voter.isMNAVotingOpen() when voting is open
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: November 28, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. National election status = 'start'
3. Voter logged in

Test Data:

- Election status: start

Test Scenario: Verify method correctly identifies MNA voting is open.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Set election status to 'start'	Status updated in DB	As Expected	Pass
2	Call isMNAVotingOpen()	Returns true	As Expected	Pass
3	Verify query execution	No errors	As Expected	Pass

Test Case ID: UNIT_041

Test Case Description: Test Voter.isMPAVotingOpen() for correct province
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 1, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. Local election started for "Punjab"
3. Voter from Punjab logged in

Test Data:

- Voter province: Punjab
- Election status: start for Punjab

Test Scenario: Verify method correctly identifies MPA voting is open for voter's province.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Start local election for Punjab	Status updated	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
2	Login voter from Punjab	Login successful	As Expected	Pass
3	Call isMPAVotingOpen()	Returns true	As Expected	Pass

Test Case ID: UNIT_042

Test Case Description: Test Voter.isMPAVotingOpen() for different province
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 1, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. Local election started for "Punjab"
3. Voter from Sindh logged in

Test Data:

- Voter province: Sindh
- Election status: start for Punjab only

Test Scenario: Verify method correctly identifies MPA voting is closed for voter's province.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Start local election for Punjab	Status updated	As Expected	Pass
2	Login voter from Sindh	Login successful	As Expected	Pass
3	Call isMPAVotingOpen()	Returns false	As Expected	Pass

Test Case ID: UNIT_043

Test Case Description: Test LocalElection.countWinner() functionality
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 5, 2025
Test Case Status: Pass

Prerequisites:

1. Database connected
2. MPA votes recorded for Punjab
3. Clear winner exists

Test Data:

- Province: Punjab
- Winner: PartyA with most votes

Test Scenario: Verify countWinner identifies Chief Minister correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Populate MPA votes for Punjab	Test data inserted	As Expected	Pass
2	Create LocalElection("Punjab")	Object created	As Expected	Pass
3	Call countWinner()	"Chief Minister from party: PartyA"	As Expected	Pass
4	Verify return value	Returns 0	As Expected	Pass

Test Case ID: UNIT_044

Test Case Description: Test Admin.viewResults() for local election
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 8, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Local election data exists for KPK
3. Database connected

Test Data:

- Province: KPK
- Multiple MPA candidates with votes

Test Scenario: Verify admin can view local election results sorted by votes.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "View Results" option	Options displayed	As Expected	Pass
2	Select "District (Local) Results"	Province prompt shown	As Expected	Pass
3	Enter "KPK"	Results displayed, sorted by votes DESC	As Expected	Pass
4	Verify data format	ID, Name, Party, Votes shown	As Expected	Pass

---

Test Case ID: UNIT_045

Test Case Description: Test Admin.viewResults() for national election
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 8, 2025
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. National election data exists for Balochistan
3. Database connected

Test Data:

- Province: Balochistan
- Multiple MNA candidates with votes

Test Scenario: Verify admin can view national election results and PM selection.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "View Results" option	Options displayed	As Expected	Pass
2	Select "Provincial (National) Results"	Province prompt shown	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Enter "Balochistan"	Results displayed, countWinner() called	As Expected	Pass
4	Verify PM selection logic	Prime Minister determination shown	As Expected	Pass

---

Test Case ID: UNIT_046

Test Case Description: Test Voter.viewResults() functionality
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 10, 2025
Test Case Status: Pass

Prerequisites:

1. Voter from Lahore, Punjab logged in
2. Election results exist
3. Database connected

Test Data:

- Voter province: Punjab
- Voter district: Lahore

Test Scenario: Verify voter can view election results for their region.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as voter from Lahore	Login successful	As Expected	Pass
2	Select "View Results" option	Results query executed	As Expected	Pass
3	View displayed results	Shows candidates from Punjab or Lahore	As Expected	Pass
4	Verify sorting	Results sorted by votes DESC	As Expected	Pass

---

Test Case ID: UNIT_047

Test Case Description: Test Voter double-voting prevention for MNA
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 12, 2025
Test Case Status: Fail

Prerequisites:

1. Voter logged in
2. MNA voting open
3. Voter has already voted for MNA

Test Data:

- Voter CNIC: 3520212345678
- has_voted_mna: 1

Test Scenario: Verify voter cannot vote twice for MNA.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Voter casts first MNA vote	Vote recorded, has_voted_mna=1	As Expected	Pass
2	Access vote menu again	"Vote for MNA" option hidden	Option still visible	Fail
3	Attempt second vote	Should be blocked	Not blocked properly	Fail

---

Test Case ID: UNIT_048

Test Case Description: Test Voter double-voting prevention for MNA (Fixed)
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: December 15, 2025
Test Case Status: Pass

Prerequisites:

1. Voter logged in
2. MNA voting open

3. Voter has already voted for MNA
4. Fixed voting logic implemented

Test Data:

- Voter CNIC: 3520212345678
- has_voted_mna: 1

Test Scenario: Verify voter cannot vote twice for MNA after fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Voter casts first MNA vote	Vote recorded, has_voted_mna=1	As Expected	Pass
2	Access vote menu again	Menu refreshes with updated state	As Expected	Pass
3	Check MNA option visibility	Option not displayed	As Expected	Pass
4	Verify message if attempted	"Cannot vote for MNA at this time"	As Expected	Pass

---

Test Case ID: UNIT_049

Test Case Description: Test Voter double-voting prevention for MPA
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.1
Date Tested: December 15, 2025
Test Case Status: Pass

Prerequisites:

1. Voter logged in
2. MPA voting open
3. Voter has already voted for MPA

Test Data:

- Voter CNIC: 3520212345678
- has_voted_mpa: 1

Test Scenario: Verify voter cannot vote twice for MPA.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Voter casts first MPA vote	Vote recorded, has_voted_mpa=1	As Expected	Pass
2	Access vote menu again	Menu refreshes with updated state	As Expected	Pass
3	Check MPA option visibility	Option not displayed	As Expected	Pass
4	Verify message if attempted	"Cannot vote for MPA at this time"	As Expected	Pass

---

Test Case ID: UNIT_050

Test Case Description: Test Voter.voteFor() with valid candidate selection
Created By: Abdul Muiz
Reviewed By: Usman
Version: 1.0
Date Tested: December 18, 2025
Test Case Status: Pass

Prerequisites:

1. Voter from Lahore, Punjab logged in
2. MNA voting open
3. Has not voted for MNA yet
4. Candidate ID 7 exists for Punjab

Test Data:

- Vote type: MNA
- Candidate ID: 7
- Province: Punjab

Test Scenario: Verify voter can successfully cast vote for valid MNA candidate.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Vote for MNA" option	Candidates list displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
2	View available candidates	Shows MNA candidates from Punjab	As Expected	Pass
3	Enter candidate ID 7	ID validated against displayed list	As Expected	Pass
4	Submit vote	"Your vote for MNA has been cast. Thank you!"	As Expected	Pass
5	Verify vote count increment	Candidate 7 votes += 1	As Expected	Pass
6	Verify voter status update	has_voted_mna = 1	As Expected	Pass

---

INTEGRATION LEVEL TESTING

Tester: Usman
Project Duration: September 15, 2025 - May 15, 2026

Test Case ID: INT_001

Test Case Description: Test CNIC Validation Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 8, 2026
Test Case Status: Pass

Prerequisites:

1. Voter registration form accessible
2. Validator methods available
3. Database connected

Test Data:

- Valid CNIC: 3520212345678
- Invalid CNICs: 123, ABC1234567890, 35202-1234567-8

Test Scenario: Verify CNIC validation integrates with registration process correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter valid CNIC "3520212345678"	Validator.isNumeric() returns true, length check passes	As Expected	Pass
2	Enter "123"	Length validation fails, error displayed	As Expected	Pass
3	Enter "ABC1234567890"	isNumeric() returns false, error displayed	As Expected	Pass
4	Enter "35202-1234567-8"	isNumeric() returns false due to dashes	As Expected	Pass

Test Case ID: INT_002

Test Case Description: Test Voter Registration to Database Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 10, 2026
Test Case Status: Pass

Prerequisites:

1. Database connected
2. Voter registration form
3. Validator class functional

Test Data:

- CNIC: 3740512345678
- Name: "Sara Ahmed"
- Province: "Sindh"
- District: "Karachi"
- Password: "secure123"

Test Scenario: Verify complete voter registration flow from input validation to database insertion.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter valid registration data	All validations pass	As Expected	Pass
2	Call registerVoter()	SQL INSERT query generated	As Expected	Pass
3	Execute database insertion	Record inserted, ID generated	As Expected	Pass
4	Verify voter data in DB	All fields match input data	As Expected	Pass
5	Check default voting flags	has_voted_mna=0, has_voted_mpa=0	As Expected	Pass

---

Test Case ID: INT_003

Test Case Description: Test Duplicate CNIC Detection Across Components
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 12, 2026
Test Case Status: Pass

Prerequisites:

1. Existing voter with CNIC 3520212345678
2. Database connected
3. Registration form accessible

Test Data:

- Duplicate CNIC: 3520212345678
- New name: "Different Person"

Test Scenario: Verify system prevents duplicate CNIC across Voter class and Database.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter duplicate CNIC	CNIC passes format validation	As Expected	Pass
2	System queries database	SELECT query executed for CNIC	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Database returns existing record	Record found with matching CNIC	As Expected	Pass
4	Registration blocked	"This CNIC is already registered" displayed	As Expected	Pass
5	Verify no new record created	Database count unchanged	As Expected	Pass

---

Test Case ID: INT_004

Test Case Description: Test Voter Login Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 15, 2026
Test Case Status: Fail

Prerequisites:

1. Registered voter in database
2. Login interface accessible
3. Database connected

Test Data:

- CNIC: 3520212345678
- Password: "voter123"

Test Scenario: Verify voter authentication integrates correctly between User class, Voter class, and Database.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter CNIC and password	Input accepted	As Expected	Pass
2	Call Voter.login()	SQL SELECT query with credentials	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Database authentication	Record found, data retrieved	As Expected	Pass
4	Set voter properties	name, province, district, voting flags loaded	Properties not fully loaded	Fail
5	Update session state	isLoggedIn = true	isLoggedIn set but flags missing	Fail

---

Test Case ID: INT_005

Test Case Description: Test Voter Login Integration (Fixed)
Created By: Usman
Reviewed By: Umair
Version: 1.1
Date Tested: January 17, 2026
Test Case Status: Pass

Prerequisites:

1. Registered voter in database
2. Login interface accessible
3. Database connected with fixed property loading

Test Data:

- CNIC: 3520212345678
- Password: "voter123"

Test Scenario: Verify voter authentication integrates correctly after property loading fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter CNIC and password	Input accepted	As Expected	Pass
2	Call Voter.login()	SQL SELECT query executed	As Expected	Pass
3	Database authentication	Record found, all fields retrieved	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
4	Set voter properties	name, province, district, has_voted_mna, has_voted_mpa loaded	As Expected	Pass
5	Update session state	isLoggedIn = true, all flags set correctly	As Expected	Pass

---

Test Case ID: INT_006

Test Case Description: Test Admin Login to Menu Navigation Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 20, 2026
Test Case Status: Pass

Prerequisites:

1. Admin class instantiated
2. Main menu accessible

Test Data:

- Username: admin
- Password: admin123

Test Scenario: Verify admin authentication flow from User interface through Admin class to menu display.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Login as Admin" from main menu	Admin.login() called	As Expected	Pass
2	Enter admin credentials	Credentials validated locally	As Expected	Pass
3	Successful authentication	login() returns true	As Expected	Pass
4	Navigate to admin menu	showMenu() called automatically	As Expected	Pass
5	Display admin options	All 6 menu options visible	As Expected	Pass

Test Case ID: INT_007

Test Case Description: Test Candidate Login Type Selection Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 22, 2026
Test Case Status: Pass

Prerequisites:

1. Candidate records in database (MNA and MPA)
2. Main menu accessible
3. Database connected

Test Data:

- MNA candidate ID: 5, Password: "mna123"
- MPA candidate ID: 12, Password: "mpa123"

Test Scenario: Verify candidate type selection integrates with polymorphic login and menu display.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Login as Candidate"	Type selection prompt shown	As Expected	Pass
2	Choose MNA (option 1)	MNA object instantiated via polymorphism	As Expected	Pass
3	Login as MNA with ID 5	Database query for MNA type, login successful	As Expected	Pass
4	Access MNA menu	Candidate::showMenu() displays correctly	As Expected	Pass
5	Logout and choose MPA	MPA object instantiated	As Expected	Pass
6	Login as MPA with ID 12	Database query for MPA type, login successful	As Expected	Pass

Test Case ID: INT_008

Test Case Description: Test Election Status Check Integration for MNA Voting
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 25, 2026
Test Case Status: Pass

Prerequisites:

1. Voter logged in
2. Database connected
3. NationalElection started

Test Data:

- Election status: level='national', status='start'
- Voter has_voted_mna: 0

Test Scenario: Verify voter menu integrates with election status checks and voting availability.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin starts national election	NationalElection.start() updates DB	As Expected	Pass
2	Voter accesses menu	showMenu() called	As Expected	Pass
3	Menu checks isMNAVotingOpen()	Query election_status table	As Expected	Pass
4	Status returns true	"Vote for MNA" option displayed	As Expected	Pass
5	Voter selects vote option	voteFor("MNA") accessible	As Expected	Pass

---

Test Case ID: INT_009

Test Case Description: Test Election Status Check Integration for MPA Voting
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 25, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Punjab logged in
2. Database connected
3. LocalElection started for Punjab

Test Data:

- Election status: level='local', status='start', province='Punjab'
- Voter province: Punjab
- Voter has_voted_mpa: 0

Test Scenario: Verify province-specific election status checking integrates correctly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin starts local election for Punjab	LocalElection("Punjab").start() updates DB	As Expected	Pass
2	Punjab voter accesses menu	showMenu() called	As Expected	Pass
3	Menu checks isMPAVotingOpen()	Query with province filter	As Expected	Pass
4	Status returns true for Punjab	"Vote for MPA" option displayed	As Expected	Pass
5	Sindh voter accesses menu	isMPAVotingOpen() returns false	As Expected	Pass

---

Test Case ID: INT_010

Test Case Description: Test Complete Voting Flow Integration for MNA
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 28, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Punjab logged in
2. MNA voting open

3. MNA candidates exist for Punjab
4. Voter has_voted_mna: 0

Test Data:

- Voter province: Punjab
- Candidate ID: 7 (MNA for Punjab)
- Candidate current votes: 150

Test Scenario: Verify complete voting process integrates Voter, Database, and Candidate components.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Vote for MNA"	voteFor("MNA") called	As Expected	Pass
2	Fetch MNA candidates	Query candidates WHERE type='MNA' AND province='Punjab'	As Expected	Pass
3	Display candidates list	All Punjab MNA candidates shown with ID, name, party	As Expected	Pass
4	Enter candidate ID 7	Validator.getIntInput() validates	As Expected	Pass
5	Validate ID against list	ID 7 exists in displayed candidates	As Expected	Pass
6	Update candidate votes	UPDATE candidates SET votes=votes+1 WHERE id=7	As Expected	Pass
7	Update voter status	UPDATE voters SET has_voted_mna=1 WHERE cnic=...	As Expected	Pass
8	Verify vote increment	Candidate 7 votes now 151	As Expected	Pass
9	Verify voter flag	Voter has_voted_mna = 1 in DB and memory	As Expected	Pass

Test Case ID: INT_011

Test Case Description: Test Complete Voting Flow Integration for MPA
Created By: Usman

Reviewed By: Umair
Version: 1.0
Date Tested: January 28, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Lahore, Punjab logged in
2. MPA voting open for Punjab
3. MPA candidates exist for Lahore
4. Voter has_voted_mpa: 0

Test Data:

- Voter district: Lahore
- Candidate ID: 15 (MPA for Lahore)
- Candidate current votes: 89

Test Scenario: Verify complete voting process for district-level MPA candidates.

Step #	Step Details	Expected Results	Actual Results	Status
1	Select **Vote for MPA** option	voteFor("MPA") function is called	As Expected	Pass
2	Fetch MPA candidates	Candidates fetched where type = **MPA** and district = **Lahore**	As Expected	Pass
3	Display candidates list	All Lahore MPA candidates are displayed	As Expected	Pass
4	Enter candidate ID **15**	Candidate ID input validated	As Expected	Pass
5	Validate ID against district list	Candidate ID 15 exists in Lahore list	As Expected	Pass
6	Update candidate votes	Candidate vote count incremented	As Expected	Pass
7	Update voter status	has_voted_mpa = 1 is set	As Expected	Pass
8	Verify district filtering	Only Lahore candidates were available	As Expected	Pass

Test Case ID: INT_012

Test Case Description: Test Admin Add Candidate to Database Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: January 30, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Database connected
3. Validator class available

Test Data:

- Name: "Zainab Hassan"
- Party: "PartyB"
- Type: "MPA"
- Province: "Sindh"
- District: "Karachi"
- Age: 38

Test Scenario: Verify candidate addition integrates input validation, Admin class, and Database insertion.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Add Candidate" from admin menu	addCandidate() called	As Expected	Pass
2	Enter candidate details	All inputs accepted	As Expected	Pass
3	Validate age input	Validator.getIntInput() processes age	As Expected	Pass
4	Generate SQL INSERT query	Query string built with all fields	As Expected	Pass
5	Execute database insertion	mysql_query() returns 0 (success)	As Expected	Pass
6	Verify candidate in DB	Record exists with votes=0	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
7	Display success message	"Candidate added successfully"	As Expected	Pass

---

Test Case ID: INT_013

Test Case Description: Test Admin Remove Candidate Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 2, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Candidate with ID 20 exists
3. Database connected

Test Data:

- Candidate ID: 20

Test Scenario: Verify candidate removal integrates validation, Admin class, and Database deletion.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Remove Candidate"	removeCandidate() called	As Expected	Pass
2	Enter candidate ID 20	Validator.getIntInput() validates	As Expected	Pass
3	Generate DELETE query	Query string: DELETE FROM candidates WHERE id=20	As Expected	Pass
4	Execute deletion	mysql_query() returns 0	As Expected	Pass
5	Verify removal from DB	SELECT returns 0 rows for ID 20	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
6	Display confirmation	"Candidate removed successfully"	As Expected	Pass

---

Test Case ID: INT_014

Test Case Description: Test Candidate View Details Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 5, 2026
Test Case Status: Pass

Prerequisites:

1. MNA candidate logged in with ID 5
2. Database connected
3. Complete candidate record exists

Test Data:

- Candidate ID: 5
- Name: "Ahmed Khan"
- Party: "PartyA"
- Province: "KPK"
- District: "Peshawar"
- Age: 45

Test Scenario: Verify candidate details retrieval integrates Candidate class and Database query.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as MNA candidate ID 5	Authentication successful, id=5 stored	As Expected	Pass
2	Select "View My Details"	viewDetails() called	As Expected	Pass
3	Execute SELECT query	Query with id=5 executed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
4	Retrieve result set	MYSQL_RES populated	As Expected	Pass
5	Fetch row data	MYSQL_ROW contains all fields	As Expected	Pass
6	Display formatted output	All details shown correctly	As Expected	Pass
7	Free result set	mysql_free_result() called	As Expected	Pass

---

Test Case ID: INT_015

Test Case Description: Test Candidate View Votes Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 5, 2026
Test Case Status: Pass

Prerequisites:

1. MPA candidate logged in with ID 12
2. Database connected
3. Vote count exists for candidate

Test Data:

- Candidate ID: 12
- Current votes: 425

Test Scenario: Verify vote count retrieval integrates Candidate class and real-time database data.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as MPA candidate ID 12	id=12 stored	As Expected	Pass
2	Select "View My Votes"	viewVotes() called	As Expected	Pass
3	Query vote count	SELECT votes FROM candidates WHERE id=12	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
4	Display current count	"Your current vote count: 425"	As Expected	Pass
5	Cast additional vote	Another voter votes for candidate 12	As Expected	Pass
6	View votes again	Updated count 426 displayed	As Expected	Pass

Test Case ID: INT_016

Test Case Description: Test Election Start/Stop Admin Control Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 8, 2026
Test Case Status: Fail

Prerequisites:

1. Admin logged in
2. Database connected
3. Election initially stopped

Test Data:

- Election type: National
- Initial status: stop

Test Scenario: Verify election control integrates Admin actions with Election classes and Database.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Start Voting"	startVoting() called	As Expected	Pass
2	Choose National Election	NationalElection object created	As Expected	Pass
3	Execute start()	Database updated with status='start'	As Expected	Pass
4	Verify voters can access voting	isMNAVotingOpen() returns true	Returns false	Fail

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
5	Check database status	status='start' in election_status	Status correctly set but check fails	Fail

Test Case ID: INT_017

Test Case Description: Test Election Start/Stop Admin Control Integration (Fixed)
Created By: Usman
Reviewed By: Umair
Version: 1.1
Date Tested: February 10, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Database connected with synchronized status checks
3. Election initially stopped

Test Data:

- Election type: National
- Initial status: stop

Test Scenario: Verify election control integrates correctly after status synchronization fix.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Start Voting"	startVoting() called	As Expected	Pass
2	Choose National Election	NationalElection instantiated	As Expected	Pass
3	Execute start()	status='start', level='national' in DB	As Expected	Pass
4	Voter checks voting availability	isMNAVotingOpen() queries latest status	As Expected	Pass
5	Verify voting access	Returns true, voting allowed	As Expected	Pass
6	Admin stops voting	stopVoting() called, status='stop'	As Expected	Pass
7	Voter checks again	isMNAVotingOpen() returns false	As Expected	Pass

Test Case ID: INT_018

Test Case Description: Test Results Viewing Integration for Voters
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 12, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Karachi, Sindh logged in
2. Multiple candidates with votes exist
3. Database connected

Test Data:

- Voter province: Sindh
- Voter district: Karachi
- Multiple MNA and MPA candidates

Test Scenario: Verify results display integrates voter location filtering with database queries.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as Karachi voter	province='Sindh', district='Karachi' loaded	As Expected	Pass
2	Select "View Results"	viewResults() called	As Expected	Pass
3	Generate filtered query	WHERE province='Sindh' OR district='Karachi'	As Expected	Pass
4	Execute query	Results retrieved	As Expected	Pass
5	Display candidates	Shows both MNA (Sindh) and MPA (Karachi) candidates	As Expected	Pass
6	Verify sorting	Ordered by votes DESC	As Expected	Pass
7	Check other provinces excluded	No candidates from Punjab or KPK shown	As Expected	Pass

Test Case ID: INT_019

Test Case Description: Test Admin Results Viewing with Winner Calculation Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 15, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. National election completed with votes
3. Database connected

Test Data:

- Province: Punjab
- PartyA wins Punjab MNA race

Test Scenario: Verify admin results view integrates with Election winner calculation methods.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "View Results"	viewResults() called	As Expected	Pass
2	Choose "Provincial (National) Results"	Option selected	As Expected	Pass
3	Enter province "Punjab"	Province filter applied	As Expected	Pass
4	Query MNA results	SELECT with type='MNA' AND province='Punjab'	As Expected	Pass
5	Display results sorted	Candidates shown, ordered by votes	As Expected	Pass
6	Call countWinner()	NationalElection::countWinner() executed	As Expected	Pass
7	Calculate PM	Aggregates provincial winners across all provinces	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
8	Display PM determination	"Prime Minister from Party..." shown	As Expected	Pass

---

Test Case ID: INT_020

Test Case Description: Test Local Election Winner Calculation Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 18, 2026
Test Case Status: Pass

Prerequisites:

1. Local election completed for Balochistan
2. MPA votes recorded
3. Database connected

Test Data:

- Province: Balochistan
- Clear winner: PartyB

Test Scenario: Verify local election results integrate with Chief Minister determination.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create LocalElection("Balochistan")	Object instantiated	As Expected	Pass
2	Call countWinner()	Method executed	As Expected	Pass
3	Query MPA results	SELECT with type='MPA' AND province='Balochistan'	As Expected	Pass
4	Find maximum votes	MAX(votes) identified	As Expected	Pass
5	Determine winning party	Party with most MPA seats found	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
6	Display Chief Minister	"Chief Minister from party: PartyB"	As Expected	Pass
7	Return success code	Returns 0	As Expected	Pass

Test Case ID: INT_021

Test Case Description: Test Voter Registration to Login Flow Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 20, 2026
Test Case Status: Pass

Prerequisites:

1. Voter portal accessible
2. Database connected
3. New user CNIC not registered

Test Data:

- CNIC: 4210187654321
- Name: "Bilal Khan"
- Province: "KPK"
- District: "Swat"
- Password: "bilal2026"

Test Scenario: Verify end-to-end flow from registration through login integrates all Voter components.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Access voter portal	Menu with Sign In/Sign Up shown	As Expected	Pass
2	Select "Sign Up"	registerVoter() called	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Enter new voter details	All validations pass	As Expected	Pass
4	Submit registration	INSERT into voters table	As Expected	Pass
5	Confirm registration	"Registration successful! Please login..."	As Expected	Pass
6	Return to portal menu	Sign In option available	As Expected	Pass
7	Select "Sign In"	login() called	As Expected	Pass
8	Enter registered CNIC and password	Authentication query executed	As Expected	Pass
9	Successful login	Voter properties loaded, menu displayed	As Expected	Pass

---

Test Case ID: INT_022

Test Case Description: Test Invalid Candidate ID Validation During Voting
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 22, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Punjab logged in
2. MNA voting open
3. Valid candidate IDs: 3, 5, 7

Test Data:

- Invalid candidate ID: 99
- Voter province: Punjab

Test Scenario: Verify voting integration validates candidate ID against region-specific list.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Vote for MNA"	Candidates displayed with IDs 3, 5, 7	As Expected	Pass
2	Enter candidate ID 99	Input accepted by Validator	As Expected	Pass
3	Validate against displayed list	ID 99 not in valid ID array	As Expected	Pass
4	Reject vote attempt	"Invalid Candidate ID for your region"	As Expected	Pass
5	Verify no vote recorded	No database UPDATE executed	As Expected	Pass
6	Verify voter status unchanged	has_voted_mna remains 0	As Expected	Pass

---

Test Case ID: INT_023

Test Case Description: Test Cross-Province Candidate Filtering Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 25, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Sindh logged in
2. MNA voting open
3. Candidates exist in multiple provinces

Test Data:

- Voter province: Sindh
- Punjab MNA candidate ID: 5
- Sindh MNA candidate IDs: 9, 11, 13

Test Scenario: Verify voting integration prevents cross-province voting through proper filtering.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as Sindh voter	province='Sindh' loaded	As Expected	Pass
2	Select "Vote for MNA"	voteFor("MNA") called	As Expected	Pass
3	Generate candidate query	WHERE type='MNA' AND province='Sindh'	As Expected	Pass
4	Display filtered candidates	Only candidates 9, 11, 13 shown	As Expected	Pass
5	Attempt to vote for Punjab candidate 5	ID 5 not in displayed list	As Expected	Pass
6	Enter ID 5	Validation against Sindh list fails	As Expected	Pass
7	Display error	"Invalid Candidate ID for your region"	As Expected	Pass

---

Test Case ID: INT_024

Test Case Description: Test Cross-District Candidate Filtering for MPA
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 25, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from Multan, Punjab logged in
2. MPA voting open for Punjab
3. Candidates exist in multiple districts

Test Data:

- Voter district: Multan
- Lahore MPA candidate ID: 15
- Multan MPA candidate IDs: 20, 22

Test Scenario: Verify MPA voting integration enforces district-level filtering.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as Multan voter	district='Multan' loaded	As Expected	Pass
2	Select "Vote for MPA"	voteFor("MPA") called	As Expected	Pass
3	Generate candidate query	WHERE type='MPA' AND district='Multan'	As Expected	Pass
4	Display filtered candidates	Only candidates 20, 22 shown	As Expected	Pass
5	Attempt to vote for Lahore candidate 15	ID not in valid list	As Expected	Pass
6	Validation fails	"Invalid Candidate ID for your region"	As Expected	Pass
7	Verify Lahore candidates excluded	No Lahore candidates accessible	As Expected	Pass

---

Test Case ID: INT_025

Test Case Description: Test Database Connection Error Handling Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: February 28, 2026
Test Case Status: Fail

Prerequisites:

1. MySQL server stopped
2. Application attempting to start

Test Data:

- Database host: localhost
- Expected connection failure

Test Scenario: Verify application handles database connection failure across all integrated components.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Stop MySQL server	Server not running	As Expected	Pass
2	Start application	DatabaseManager::connect() called	As Expected	Pass
3	Connection attempt fails	mysql_real_connect() returns NULL	As Expected	Pass
4	Error handling	Graceful error message	Program calls exit(1) - abrupt termination	Fail
5	Verify no further execution	Application stops safely	Application crashes ungracefully	Fail

---

Test Case ID: INT_026

Test Case Description: Test Database Connection Error Handling Integration (Improved)
Created By: Usman
Reviewed By: Umair
Version: 1.1
Date Tested: March 2, 2026
Test Case Status: Pass

Prerequisites:

1. MySQL server stopped
2. Application with improved error handling

Test Data:

- Database host: localhost
- Expected connection failure

Test Scenario: Verify improved database error handling integrates properly across components.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Stop MySQL server	Server not running	As Expected	Pass
2	Start application	DatabaseManager::connect() called	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
3	Connection attempt fails	mysql_real_connect() returns NULL	As Expected	Pass
4	Display error message	"Connection to database failed: ..." shown	As Expected	Pass
5	Log error details	mysql_error() details captured	As Expected	Pass
6	Graceful termination	Clean exit with error code	As Expected	Pass

Test Case ID: INT_027

Test Case Description: Test Menu Navigation Between User Types
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 5, 2026
Test Case Status: Pass

Prerequisites:

1. Main menu accessible
2. All user types configured in database

Test Data:

- Admin credentials: admin/admin123
- MNA ID: 5
- Voter CNIC: 3520212345678

Test Scenario: Verify main menu integrates seamlessly with polymorphic user type selection.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Display main menu	4 options shown	As Expected	Pass
2	Select "Login as Admin"	Admin object created	As Expected	Pass
3	Login successfully	Admin menu displayed	As Expected	Pass
4	Logout to main menu	Return to main menu	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
5	Select "Login as Candidate"	Candidate type prompt shown	As Expected	Pass
6	Choose MNA, login	MNA-specific menu shown	As Expected	Pass
7	Logout to main menu	Return successful	As Expected	Pass
8	Select "Voter" option	Voter Sign In/Sign Up menu shown	As Expected	Pass
9	Login as voter	Voter menu with voting options shown	As Expected	Pass

---

Test Case ID: INT_028

Test Case Description: Test Voting Status Flags Integration After Vote
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 8, 2026
Test Case Status: Pass

Prerequisites:

1. Voter logged in
2. Both MNA and MPA voting open
3. Voter has not voted yet

Test Data:

- Voter CNIC: 3520212345678
- MNA candidate ID: 7
- MPA candidate ID: 15

Test Scenario: Verify voting status flags integrate correctly across vote operations and menu display.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Initial menu display	Both "Vote for MNA" and "Vote for MPA" visible	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
2	Vote for MNA candidate 7	Vote recorded, has_voted_mna=1 in DB	As Expected	Pass
3	Update memory flag	hasVotedMNA = true in Voter object	As Expected	Pass
4	Return to menu	Menu regenerated	As Expected	Pass
5	Check MNA option	"Vote for MNA" no longer displayed	As Expected	Pass
6	Check MPA option	"Vote for MPA" still visible	As Expected	Pass
7	Vote for MPA candidate 15	Vote recorded, has_voted_mpa=1 in DB	As Expected	Pass
8	Update memory flag	hasVotedMPA = true	As Expected	Pass
9	Return to menu	Neither voting option visible	As Expected	Pass

---

Test Case ID: INT_029

Test Case Description: Test Multi-Component Data Flow for Candidate Addition
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 10, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Database connected
3. Validators functional

Test Data:

- Name: "Ayesha Malik"
- Party: "PartyA"
- Type: "MNA"
- Province: "Punjab"
- District: "Faisalabad"
- Age: 42

- Password: "candidate2026"

Test Scenario: Verify data flows correctly from Admin input through validation to database storage, then to candidate login capability.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin selects "Add Candidate"	addCandidate() method called	As Expected	Pass
2	Enter all candidate details	Input streams process data	As Expected	Pass
3	Validate age with Validator	getIntInput(42) returns true, valid	As Expected	Pass
4	Build SQL INSERT query	Query string constructed with all fields	As Expected	Pass
5	Execute database insertion	Record inserted, auto-increment ID assigned	As Expected	Pass
6	Retrieve new candidate ID	ID = 25 returned	As Expected	Pass
7	Logout from admin	Return to main menu	As Expected	Pass
8	Login as MNA candidate	Select candidate type = MNA	As Expected	Pass
9	Use new ID 25 and password	Authentication successful	As Expected	Pass
10	Access candidate menu	MNA menu displayed with new candidate data	As Expected	Pass

---

Test Case ID: INT_030

Test Case Description: Test Election Lifecycle Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 12, 2026
Test Case Status: Pass

Prerequisites:

1. Admin access
2. Registered voters
3. Registered candidates
4. Database connected

Test Data:

- Province: KPK
- Local election

Test Scenario: Verify complete election lifecycle integrates admin control, voter participation, and results calculation.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin starts local election for KPK	LocalElection("KPK").start(), status='start'	As Expected	Pass
2	Voter 1 from KPK logs in	Login successful	As Expected	Pass
3	Check voting availability	isMPAVotingOpen() returns true	As Expected	Pass
4	Voter 1 casts MPA vote	Vote recorded, has_voted_mpa=1	As Expected	Pass
5	Voter 2 from KPK logs in	Login successful	As Expected	Pass
6	Voter 2 casts MPA vote	Vote recorded	As Expected	Pass
7	Admin views partial results	Results displayed with current vote counts	As Expected	Pass
8	Multiple additional voters vote	Votes accumulate	As Expected	Pass
9	Admin stops election	LocalElection("KPK").end(), status='stop'	As Expected	Pass
10	New voter attempts to vote	isMPAVotingOpen() returns false, cannot vote	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
11	Admin views final results	Final tallies shown	As Expected	Pass
12	Call countWinner()	"Chief Minister from party: ..." determined	As Expected	Pass

---

Test Case ID: INT_031

Test Case Description: Test National Election Prime Minister Determination Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 15, 2026
Test Case Status: Pass

Prerequisites:

1. National election completed
2. MNA votes recorded across all provinces
3. Admin access

Test Data:

- Punjab: PartyA wins
- Sindh: PartyB wins
- KPK: PartyA wins
- Balochistan: PartyA wins

Test Scenario: Verify PM determination integrates provincial MNA results across the nation.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin selects "View Results"	viewResults() called	As Expected	Pass
2	Choose "Provincial (National) Results"	Option selected	As Expected	Pass
3	Enter "Punjab"	Punjab MNA results displayed	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
4	System calls countWinner()	NationalElection::countWinner() executed	As Expected	Pass
5	Query all provinces	Iterates through Punjab, Sindh, KPK, Balochistan	As Expected	Pass
6	Determine provincial winners	PartyA: 3 provinces, PartyB: 1 province	As Expected	Pass
7	Calculate national winner	PartyA has majority	As Expected	Pass
8	Display PM determination	"Prime Minister from PartyA"	As Expected	Pass
9	Return success code	Returns 0	As Expected	Pass

Test Case ID: INT_032

Test Case Description: Test Concurrent Voting Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 18, 2026
Test Case Status: Pass

Prerequisites:

1. Both national and local elections open
2. Voter from Lahore, Punjab logged in
3. Both MNA and MPA candidates available

Test Data:

- Voter location: Lahore, Punjab
- MNA candidate ID: 5
- MPA candidate ID: 15

Test Scenario: Verify voter can participate in both elections simultaneously with proper integration.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin starts national election	NationalElection.start()	As Expected	Pass
2	Admin starts local election for Punjab	LocalElection("Punjab").start()	As Expected	Pass
3	Voter logs in	Both elections accessible	As Expected	Pass
4	Check menu options	Both "Vote for MNA" and "Vote for MPA" visible	As Expected	Pass
5	Vote for MNA candidate 5	Vote recorded, has_voted_mna=1	As Expected	Pass
6	Return to menu	"Vote for MNA" hidden, "Vote for MPA" still visible	As Expected	Pass
7	Vote for MPA candidate 15	Vote recorded, has_voted_mpa=1	As Expected	Pass
8	Return to menu	Both voting options hidden	As Expected	Pass
9	View results	Shows both MNA and MPA results for region	As Expected	Pass

---

Test Case ID: INT_033

Test Case Description: Test Password Authentication Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 20, 2026
Test Case Status: Pass

Prerequisites:

1. Candidate registered in database
2. Login interface accessible
3. Database connected

Test Data:

- Candidate ID: 5
- Correct password: "mna123"
- Incorrect password: "wrongpass"

Test Scenario: Verify password authentication integrates correctly between Candidate login and Database validation.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "Login as Candidate"	Type selection shown	As Expected	Pass
2	Choose MNA type	MNA object created	As Expected	Pass
3	Enter ID 5 and wrong password	Credentials accepted for processing	As Expected	Pass
4	Execute authentication query	SELECT with id=5 AND password='wrongpass'	As Expected	Pass
5	Query returns no rows	mysql_num_rows(res) == 0	As Expected	Pass
6	Display error message	"Invalid ID or password. Please try again."	As Expected	Pass
7	Retry with correct password	Enter "mna123"	As Expected	Pass
8	Execute authentication query	SELECT with id=5 AND password='mna123'	As Expected	Pass
9	Query returns 1 row	Candidate record found	As Expected	Pass
10	Load candidate data	username set from database	As Expected	Pass
11	Display success	"Login successful. Welcome, Ahmed Khan!"	As Expected	Pass

---

Test Case ID: INT_034

Test Case Description: Test Empty Candidate List Handling Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0

Date Tested: March 22, 2026
Test Case Status: Pass

Prerequisites:

1. Voter from remote district logged in
2. MPA voting open
3. No MPA candidates registered for voter's district

Test Data:

- Voter district: Ziarat (small district with no candidates)
- MPA candidates: 0

Test Scenario: Verify system handles empty candidate lists gracefully across Voter and Database integration.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Login as voter from Ziarat	Login successful	As Expected	Pass
2	Check MPA voting status	isMPAVotingOpen() returns true	As Expected	Pass
3	Select "Vote for MPA"	voteFor("MPA") called	As Expected	Pass
4	Query MPA candidates for Ziarat	SELECT ... WHERE district='Ziarat'	As Expected	Pass
5	Retrieve result set	MYSQL_RES returned	As Expected	Pass
6	Count candidates	idx = 0 (no candidates)	As Expected	Pass
7	Display appropriate message	"No candidates available for your region."	As Expected	Pass
8	Free result set	mysql_free_result() called	As Expected	Pass
9	Return to menu	Menu displayed without crash	As Expected	Pass

---

Test Case ID: INT_035

Test Case Description: Test Result Sorting Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0

Date Tested: March 25, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Multiple candidates with varying vote counts
3. Database connected

Test Data:

- Province: Punjab
- Candidate A: 1250 votes
- Candidate B: 2100 votes
- Candidate C: 890 votes
- Candidate D: 1800 votes

Test Scenario: Verify results display integrates proper SQL sorting across Admin view and Database query.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Select "View Results"	viewResults() called	As Expected	Pass
2	Choose "District (Local) Results"	Option selected	As Expected	Pass
3	Enter "Punjab"	Province filter applied	As Expected	Pass
4	Execute query with ORDER BY	Query includes "ORDER BY votes DESC"	As Expected	Pass
5	Retrieve sorted results	Result set ordered correctly	As Expected	Pass
6	Display first result	Candidate B (2100 votes) shown first	As Expected	Pass
7	Display second result	Candidate D (1800 votes) shown second	As Expected	Pass
8	Display third result	Candidate A (1250 votes) shown third	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
9	Display fourth result	Candidate C (890 votes) shown fourth	As Expected	Pass
10	Verify descending order	Highest votes to lowest votes confirmed	As Expected	Pass

---

Test Case ID: INT_036

Test Case Description: Test Validator Input Integration Across Multiple Forms
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 28, 2026
Test Case Status: Pass

Prerequisites:

1. Multiple forms requiring validation
2. Validator class available

Test Data:

- Admin candidate addition form
- Voter registration form
- Voting selection input

Test Scenario: Verify Validator class integrates consistently across different form inputs throughout the application.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin enters candidate age "45"	Validator.getIntInput() validates successfully	As Expected	Pass
2	Admin enters invalid age "abc"	Validator rejects, displays error	As Expected	Pass
3	Voter registers with CNIC "3520212345678"	Validator.isNumeric() and length check pass	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
4	Voter enters CNIC "35202-1234-5678"	Validator.isNumeric() rejects due to dashes	As Expected	Pass
5	Voter selects candidate ID "7"	Validator.getIntInput() processes successfully	As Expected	Pass
6	Voter enters candidate ID "xyz"	Validator rejects non-numeric input	As Expected	Pass
7	Admin enters menu choice "3"	Validator processes valid menu option	As Expected	Pass
8	Admin enters menu choice "invalid"	Validator rejects, prompts retry	As Expected	Pass

---

Test Case ID: INT_037

Test Case Description: Test Session State Management Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: March 30, 2026
Test Case Status: Pass

Prerequisites:

1. Voter logged in
2. Voting operations performed
3. Session data maintained

Test Data:

- Voter CNIC: 3520212345678
- Session flags: isLoggedIn, hasVotedMNA, hasVotedMPA

Test Scenario: Verify session state integrates correctly between Voter object properties and database persistent state.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Voter logs in	isLoggedIn=true, flags loaded from DB	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
2	Check initial voting flags	hasVotedMNA=false, hasVotedMPA=false	As Expected	Pass
3	Cast MNA vote	hasVotedMNA updated to true in memory	As Expected	Pass
4	Verify database update	has_voted_mna=1 in voters table	As Expected	Pass
5	Return to menu	Memory state preserved	As Expected	Pass
6	Access menu options	"Vote for MNA" hidden based on memory flag	As Expected	Pass
7	Logout	isLoggedIn=false, session cleared	As Expected	Pass
8	Login again	Flags reloaded: hasVotedMNA=true from DB	As Expected	Pass
9	Verify persistent state	MNA voting still unavailable	As Expected	Pass

Test Case ID: INT_038

Test Case Description: Test Multi-User Type Polymorphism Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 2, 2026
Test Case Status: Pass

Prerequisites:

1. All user types registered
2. Database connected
3. Polymorphic User base class functional

Test Data:

- Admin: admin/admin123
- MNA ID: 5
- MPA ID: 12

- Voter CNIC: 3520212345678

Test Scenario: Verify polymorphic User interface integrates correctly with specialized login and menu methods across all user types.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create Admin object via User* pointer	Polymorphic instantiation successful	As Expected	Pass
2	Call virtual login() method	Admin::login() executed	As Expected	Pass
3	Call virtual showMenu() method	Admin::showMenu() with 6 options displayed	As Expected	Pass
4	Create MNA via Candidate* pointer	Candidate polymorphism functional	As Expected	Pass
5	Call Candidate::login()	MNA-specific authentication	As Expected	Pass
6	Call Candidate::showMenu()	Candidate menu with View Details/Votes	As Expected	Pass
7	Call MNA::viewDetails()	Overridden method displays MNA-specific format	As Expected	Pass
8	Create Voter object	Voter inherits from User	As Expected	Pass
9	Call Voter::login()	Voter-specific Sign In/Sign Up flow	As Expected	Pass
10	Call Voter::showMenu()	Voter menu with voting/results options	As Expected	Pass

Test Case ID: INT_039

Test Case Description: Test Election Class Inheritance Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 5, 2026
Test Case Status: Pass

Prerequisites:

1. Election base class defined
2. LocalElection and NationalElection derived classes
3. Database connected

Test Data:

- Province: Sindh

Test Scenario: Verify Election class inheritance integrates virtual methods correctly across derived classes.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create Election* pointer to LocalElection	Polymorphic creation successful	As Expected	Pass
2	Call virtual start() method	LocalElection::start() executed for Sindh	As Expected	Pass
3	Verify database update	level='local', province='Sindh'	As Expected	Pass
4	Call virtual countWinner()	LocalElection::countWinner() finds Chief Minister	As Expected	Pass
5	Call virtual end() method	LocalElection::end() updates status	As Expected	Pass
6	Create Election* to NationalElection	Polymorphic switch successful	As Expected	Pass
7	Call virtual start()	NationalElection::start() for national level	As Expected	Pass
8	Verify database update	level='national', province=''	As Expected	Pass
9	Call virtual countWinner()	NationalElection::countWinner() aggregates provinces	As Expected	Pass
10	Verify polymorphic behavior	Correct method called based on object type	As Expected	Pass

Test Case ID: INT_040

Test Case Description: Test Database Query Error Handling Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 8, 2026
Test Case Status: Pass

Prerequisites:

1. Database connected
2. Intentionally malformed query
3. Error handling implemented

Test Data:

- Invalid query: Missing table name

Test Scenario: Verify database error handling integrates across all components that execute queries.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Execute malformed query in Admin.addCandidate()	mysql_query() returns non-zero	As Expected	Pass
2	Check return value	Error detected	As Expected	Pass
3	Display error message	"Error adding candidate: [error details]"	As Expected	Pass
4	Verify no partial data saved	No incomplete record in database	As Expected	Pass
5	Execute malformed query in Voter.voteFor()	Error detected before vote recorded	As Expected	Pass
6	Handle gracefully	Error message shown, return to menu	As Expected	Pass
7	Test error in results retrieval	Query error caught	As Expected	Pass
8	Display appropriate message	"Error fetching results: ..."	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
9	Verify application stability	No crashes, continues functioning	As Expected	Pass

---

Test Case ID: INT_041

Test Case Description: Test Memory Management Integration in Polymorphic Deletion
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 10, 2026
Test Case Status: Pass

Prerequisites:

1. Polymorphic user objects created
2. Virtual destructors implemented

Test Data:

- MNA and MPA candidate objects

Test Scenario: Verify proper memory cleanup integrates with polymorphic object deletion.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Create Candidate* pointer to new MNA()	Dynamic allocation successful	As Expected	Pass
2	Use MNA object (login, menu)	Object functions correctly	As Expected	Pass
3	Delete candidate pointer	Virtual destructor called	As Expected	Pass
4	Verify memory freed	No memory leaks	As Expected	Pass
5	Create Candidate* to new MPA()	Allocation successful	As Expected	Pass
6	Use MPA object	Functions correctly	As Expected	Pass
7	Delete candidate pointer	Proper cleanup via virtual destructor	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
8	Run multiple login/logout cycles	No memory accumulation	As Expected	Pass

---

Test Case ID: INT_042

Test Case Description: Test End-to-End Voting Process Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 12, 2026
Test Case Status: Pass

Prerequisites:

1. Complete system setup
2. Admin, candidates, and voters registered
3. Database connected

Test Data:

- Multiple voters
- Multiple candidates across types
- Both elections active

Test Scenario: Verify complete end-to-end voting process integrates all system components seamlessly.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin logs in	Authentication successful	As Expected	Pass
2	Admin adds candidates	Candidates stored in database	As Expected	Pass
3	Admin starts both elections	Election statuses updated	As Expected	Pass
4	Voter 1 registers	Registration completed	As Expected	Pass
5	Voter 1 logs in	Login successful, flags loaded	As Expected	Pass
6	Voter 1 votes for MNA	Vote recorded, flag updated	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
7	Voter 1 votes for MPA	Vote recorded, flag updated	As Expected	Pass
8	Voter 2 performs same process	All operations successful	As Expected	Pass
9	Candidate logs in and views votes	Current vote count displayed	As Expected	Pass
10	Admin views partial results	Results show accumulated votes	As Expected	Pass
11	Multiple voters continue voting	All votes recorded correctly	As Expected	Pass
12	Admin stops elections	Statuses updated, voting blocked	As Expected	Pass
13	Admin views final results	Complete results with winners shown	As Expected	Pass
14	Verify data integrity	All votes accounted for, no duplicates	As Expected	Pass

---

Test Case ID: INT_043

Test Case Description: Test Province-Based Election Independence Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 15, 2026
Test Case Status: Pass

Prerequisites:

1. Admin access
2. Voters from different provinces
3. Candidates in multiple provinces

Test Data:

- Punjab election: Started
- Sindh election: Stopped

- KPK election: Not started

Test Scenario: Verify provincial election independence integrates correctly with voter access controls.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin starts local election for Punjab only	status='start' for Punjab	As Expected	Pass
2	Punjab voter logs in	Login successful	As Expected	Pass
3	Check MPA voting availability	isMPAVotingOpen() returns true	As Expected	Pass
4	Punjab voter casts MPA vote	Vote successful	As Expected	Pass
5	Sindh voter logs in	Login successful	As Expected	Pass
6	Check MPA voting availability	isMPAVotingOpen() returns false	As Expected	Pass
7	Verify Sindh voter cannot vote	"Cannot vote for MPA at this time"	As Expected	Pass
8	KPK voter logs in	Login successful	As Expected	Pass
9	Check MPA voting availability	isMPAVotingOpen() returns false	As Expected	Pass
10	Start election for Sindh	Sindh election started independently	As Expected	Pass
11	Sindh voter can now vote	MPA voting accessible	As Expected	Pass
12	Punjab election remains active	Punjab status unchanged	As Expected	Pass

Test Case ID: INT_044

Test Case Description: Test Candidate-Specific Data Isolation Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 18, 2026
Test Case Status: Pass

Prerequisites:

1. Multiple candidates registered
2. Database connected
3. Candidates of same type (MNA) logged in

Test Data:

- Candidate A (ID 5): 1500 votes
- Candidate B (ID 7): 2200 votes

Test Scenario: Verify candidate data isolation integrates correctly so each candidate sees only their own information.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Candidate A logs in with ID 5	id=5 stored in object	As Expected	Pass
2	Candidate A views details	Shows Candidate A's information only	As Expected	Pass
3	Candidate A views votes	Displays 1500 votes	As Expected	Pass
4	Candidate A logs out	Session cleared	As Expected	Pass
5	Candidate B logs in with ID 7	id=7 stored in object	As Expected	Pass
6	Candidate B views details	Shows Candidate B's information only	As Expected	Pass
7	Candidate B views votes	Displays 2200 votes	As Expected	Pass
8	Verify data isolation	No cross-contamination between candidate sessions	As Expected	Pass

---

Test Case ID: INT_045

Test Case Description: Test Concurrent Admin Operations Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 20, 2026
Test Case Status: Pass

Prerequisites:

1. Admin logged in
2. Active elections
3. Database connected

Test Data:

- Various admin operations

Test Scenario: Verify multiple admin operations integrate correctly in sequence without state conflicts.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Admin starts national election	Status updated	As Expected	Pass
2	Admin adds new candidate	Candidate inserted	As Expected	Pass
3	Admin views current results	Results displayed	As Expected	Pass
4	Admin adds another candidate	Second candidate inserted	As Expected	Pass
5	Admin starts local election for Sindh	Sindh election started	As Expected	Pass
6	Admin views results again	Both election results shown	As Expected	Pass
7	Admin removes a candidate	Candidate deleted	As Expected	Pass
8	Admin views updated results	Deleted candidate not shown	As Expected	Pass
9	Admin stops national election	National status updated	As Expected	Pass
10	Verify all operations persisted	Database state reflects all changes	As Expected	Pass

---

Test Case ID: INT_046

Test Case Description: Test Input Validation Chain Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 22, 2026
Test Case Status: Pass

Prerequisites:

1. Multi-step forms requiring validation
2. Validator class functional
3. Database connected

Test Data:

- Voter registration with multiple fields

Test Scenario: Verify validation chain integrates correctly across multiple input fields in sequence.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Enter CNIC "abc123"	Validator.isNumeric() rejects	As Expected	Pass
2	Re-enter CNIC "3520212345678"	Validation passes	As Expected	Pass
3	Enter name with numbers "Ali123"	Accepted (no alphabetic-only requirement)	As Expected	Pass
4	Enter province "Punjab"	Accepted	As Expected	Pass
5	Enter district "Lahore"	Accepted	As Expected	Pass
6	Enter password "pass"	Accepted (no complexity requirement)	As Expected	Pass
7	Submit registration	All validations passed, record inserted	As Expected	Pass
8	Verify database entry	All fields stored correctly	As Expected	Pass

---

Test Case ID: INT_047

Test Case Description: Test Results Query Performance Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 25, 2026
Test Case Status: Pass

Prerequisites:

1. Large dataset (100+ candidates with votes)
2. Database connected

3. Query optimization implemented

Test Data:

- 150 candidates across all provinces
- Various vote counts

Test Scenario: Verify results retrieval integrates efficiently with database queries for large datasets.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Populate database with 150 candidates	Data inserted	As Expected	Pass
2	Admin views results for Punjab	Query executes with province filter	As Expected	Pass
3	Measure query execution time	Response < 2 seconds	As Expected	Pass
4	Verify results are sorted	ORDER BY votes DESC applied	As Expected	Pass
5	Display results	All Punjab candidates shown correctly	As Expected	Pass
6	Voter views results for Lahore	Query with district filter	As Expected	Pass
7	Measure query execution time	Response < 2 seconds	As Expected	Pass
8	Verify filtering accuracy	Only Lahore candidates displayed	As Expected	Pass

---

Test Case ID: INT_048

Test Case Description: Test Transaction-Like Voting Integrity Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: April 28, 2026
Test Case Status: Pass

Prerequisites:

1. Voter logged in
2. Voting process started

3. Database connected

Test Data:

- Voter CNIC: 3520212345678
- Candidate ID: 7

Test Scenario: Verify voting process maintains data integrity across multiple database operations.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Voter initiates vote for candidate 7	Process started	As Expected	Pass
2	Check initial candidate votes	Current count retrieved	As Expected	Pass
3	Execute UPDATE for candidate votes	Votes incremented	As Expected	Pass
4	Verify vote increment successful	mysql_query() returns 0	As Expected	Pass
5	Execute UPDATE for voter status	has_voted_mna updated	As Expected	Pass
6	Verify voter update successful	mysql_query() returns 0	As Expected	Pass
7	Verify both updates completed	Database reflects both changes	As Expected	Pass
8	Check data consistency	Vote count and voter flag match	As Expected	Pass

---

Test Case ID: INT_049

Test Case Description: Test Menu State Refresh Integration
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: May 1, 2026
Test Case Status: Pass

Prerequisites:

1. Voter logged in

2. Dynamic menu based on voting status
3. Voting operations performed

Test Data:

- Initial state: No votes cast
- After operations: Votes cast

Test Scenario: Verify menu dynamically refreshes to integrate updated voting status after operations.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Display initial menu	Both MNA and MPA options visible	As Expected	Pass
2	Cast MNA vote	Vote operation completed	As Expected	Pass
3	Return to menu	showMenu() called again	As Expected	Pass
4	Check menu state refresh	isMNAVotingOpen() called again	As Expected	Pass
5	Check hasVotedMNA flag	Value is now true	As Expected	Pass
6	Verify MNA option hidden	"Vote for MNA" not displayed	As Expected	Pass
7	Verify MPA option visible	"Vote for MPA" still displayed	As Expected	Pass
8	Cast MPA vote	Vote operation completed	As Expected	Pass
9	Return to menu	Menu refreshes	As Expected	Pass
10	Verify both options hidden	Neither voting option displayed	As Expected	Pass

---

Test Case ID: INT_050

Test Case Description: Test Complete System Integration Workflow
Created By: Usman
Reviewed By: Umair
Version: 1.0
Date Tested: May 5, 2026
Test Case Status: Pass

Prerequisites:

1. Fresh database state
2. All system components functional
3. Test data prepared

Test Data:

- Complete election scenario with multiple users

Test Scenario: Verify complete system integration from initial setup through election completion.

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
1	Initialize database connection	Connection successful	As Expected	Pass
2	Admin logs in	Authentication successful	As Expected	Pass
3	Admin adds 10 MNA candidates	All candidates inserted	As Expected	Pass
4	Admin adds 15 MPA candidates	All candidates inserted	As Expected	Pass
5	Admin starts national election	Status updated nationally	As Expected	Pass
6	Admin starts local elections for 4 provinces	All provincial elections started	As Expected	Pass
7	50 voters register	All registrations successful	As Expected	Pass
8	50 voters login and vote for MNA	All MNA votes recorded	As Expected	Pass
9	50 voters vote for MPA	All MPA votes recorded	As Expected	Pass
10	Candidates view their vote counts	All counts accurate and isolated	As Expected	Pass
11	Admin views partial results	Results displayed correctly	As Expected	Pass
12	Admin stops all elections	All statuses updated	As Expected	Pass
13	New voter attempts to register and vote	Registration works, voting blocked	As Expected	Pass

Step #	Step Details	Expected Results	Actual Results	Pass/Fail
14	Admin views final results	Winners determined correctly	As Expected	Pass
15	Verify Prime Minister selection	PM determined from provincial winners	As Expected	Pass
16	Verify Chief Ministers	CMs determined for each province	As Expected	Pass
17	Verify data integrity	All votes accounted for, no anomalies	As Expected	Pass
18	Disconnect database	Clean disconnection	As Expected	Pass