

Mobile

Test Report

Test Session Summary

|  |  |
| --- | --- |
| Start: Mon Apr 13 06:07:37 2020 End: Mon Apr 13 06:26:26 2020 Duration: 0:18:49.066000  broken: 0 (0.00%) failed: 7 (17.50%) skipped: 0 (0.00%) passed: 33 (82.50%) |  |

Test Results

# Liked tracks clicking - failed

Liked tracks clicking and check that it works

## Details

AssertionError: assert False

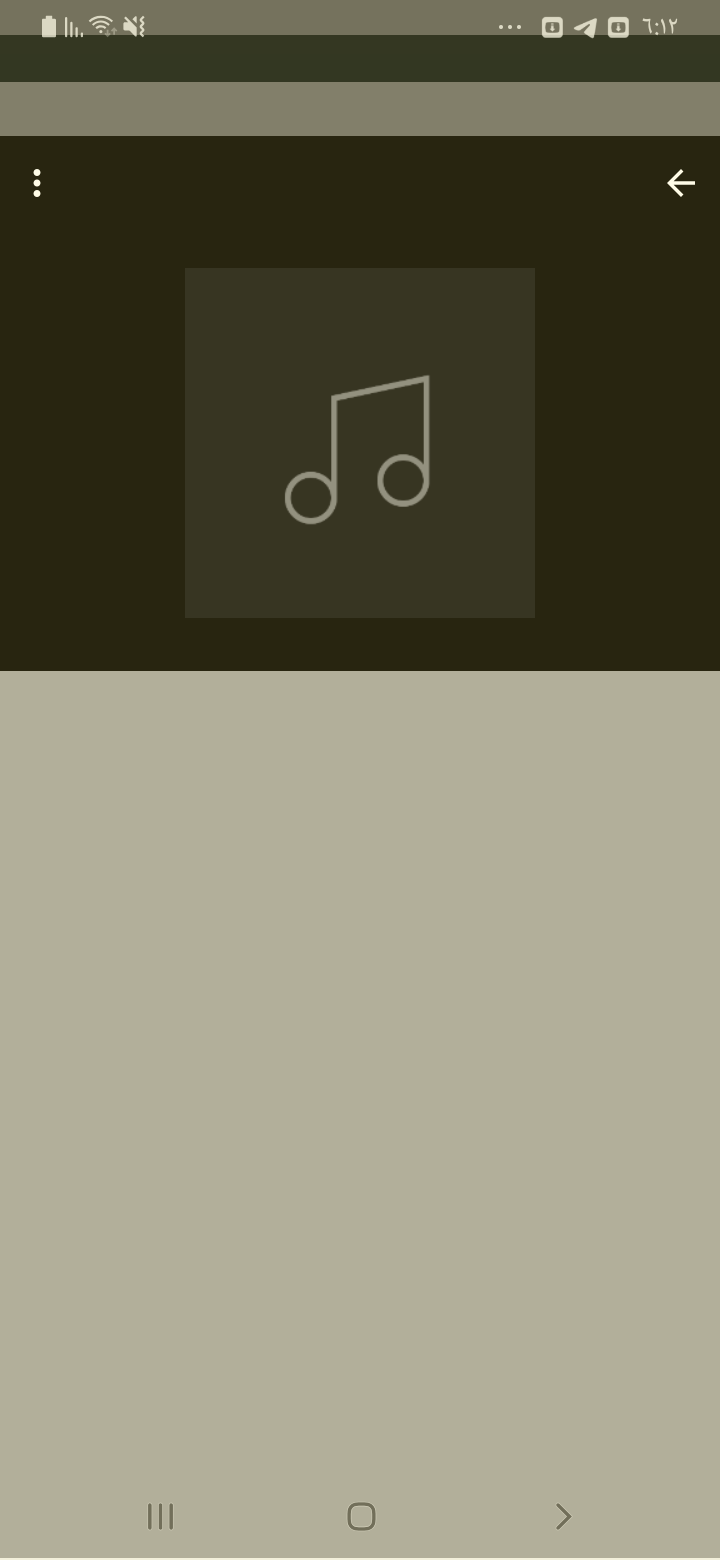
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_home.TestHome object at 0x110427970>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Home Page Tests")  @allure.sub\_suite("Liked tracks clicking")  @allure.title("Liked tracks clicking")  @allure.description("Liked tracks clicking and check that it works")  @pytest.mark.Do  @pytest.mark.HomePage  @pytest.mark.Test3  def test\_case\_3(self, setup):  """  checking the Liked tracks works  """  ap = AuthenticationPage(self.driver)  ap.click\_signin\_button()  lp = LoginPage(self.driver)  lp.do\_the\_login(Constants.correct\_credentials["email"], Constants.correct\_credentials["password"])  hp = HomePage(self.driver)  time.sleep(10)  hp.click\_liked\_tracks\_button()  if Helper.element\_exists\_by\_id(self.driver, HomePage.about\_button\_id):  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Home page test failed") > assert False E assert False  Mobile\_Testing/Tests/test\_home.py:140: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# New releases clicking - failed

New releases clicking and check that it works

## Details

AssertionError: assert False

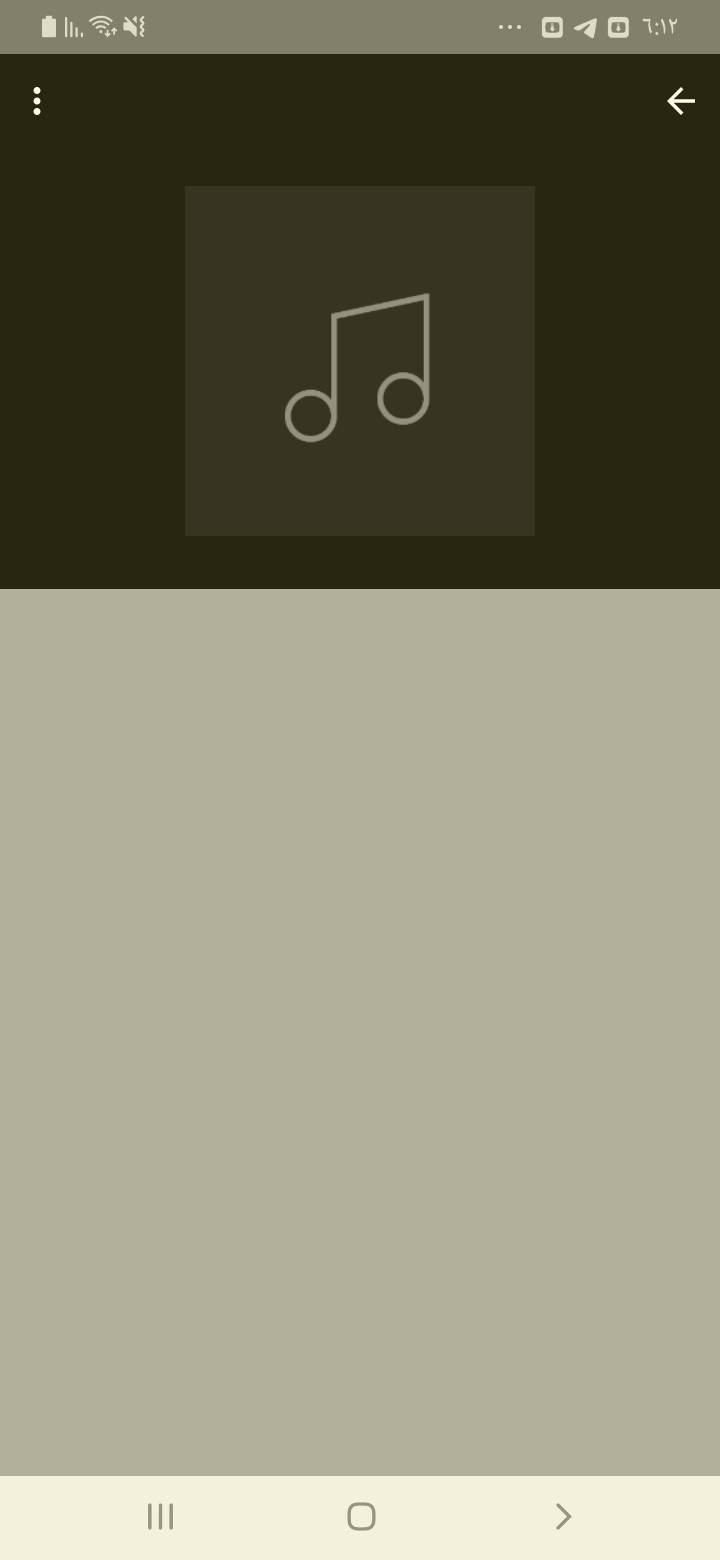
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_home.TestHome object at 0x1103adf70>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Home Page Tests")  @allure.sub\_suite("New releases clicking")  @allure.title("New releases clicking")  @allure.description("New releases clicking and check that it works")  @pytest.mark.Do  @pytest.mark.HomePage  @pytest.mark.Test2  def test\_case\_2(self, setup):  """  checking the New releases works  """  ap = AuthenticationPage(self.driver)  ap.click\_signin\_button()  lp = LoginPage(self.driver)  lp.do\_the\_login(Constants.correct\_credentials["email"], Constants.correct\_credentials["password"])  hp = HomePage(self.driver)  time.sleep(10)  hp.click\_new\_releases\_button()  if Helper.element\_exists\_by\_id(self.driver, HomePage.about\_button\_id):  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Home page test failed") > assert False E assert False  Mobile\_Testing/Tests/test\_home.py:111: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# Popular tracks clicking - failed

Popular tracks clicking and check that it works

## Details

AssertionError: assert False

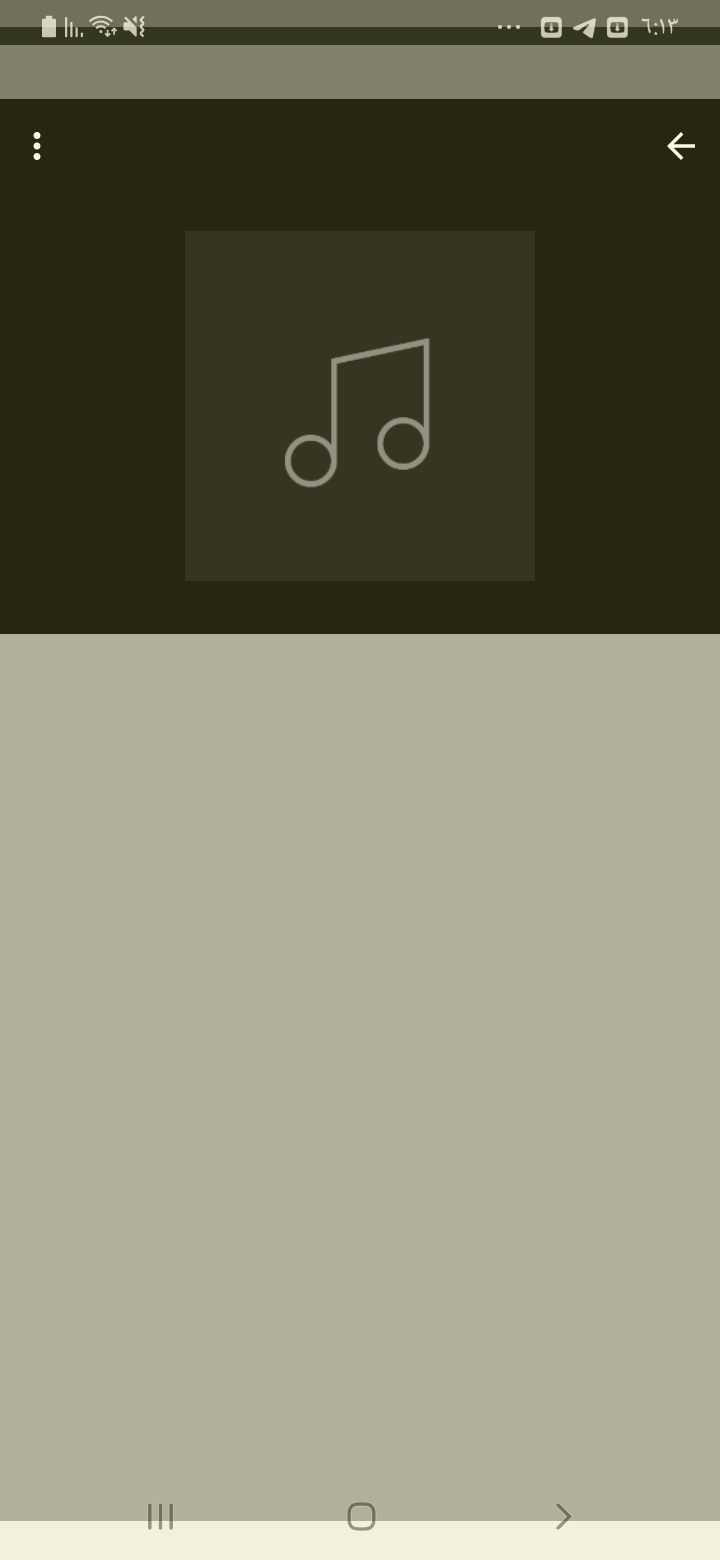
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_home.TestHome object at 0x11045dc40>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Home Page Tests")  @allure.sub\_suite("Popular tracks clicking")  @allure.title("Popular tracks clicking")  @allure.description("Popular tracks clicking and check that it works")  @pytest.mark.Do  @pytest.mark.HomePage  @pytest.mark.Test4  def test\_case\_4(self, setup):  """  checking the Popular tracks works  """  ap = AuthenticationPage(self.driver)  ap.click\_signin\_button()  lp = LoginPage(self.driver)  lp.do\_the\_login(Constants.correct\_credentials["email"], Constants.correct\_credentials["password"])  hp = HomePage(self.driver)  time.sleep(10)  hp.click\_popular\_tracks\_button()  if Helper.element\_exists\_by\_id(self.driver, HomePage.about\_button\_id):  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Home page test failed") > assert False E assert False  Mobile\_Testing/Tests/test\_home.py:169: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# Recommended tracks clicking - failed

Recommended tracks clicking and check that it works

## Details

AssertionError: assert False

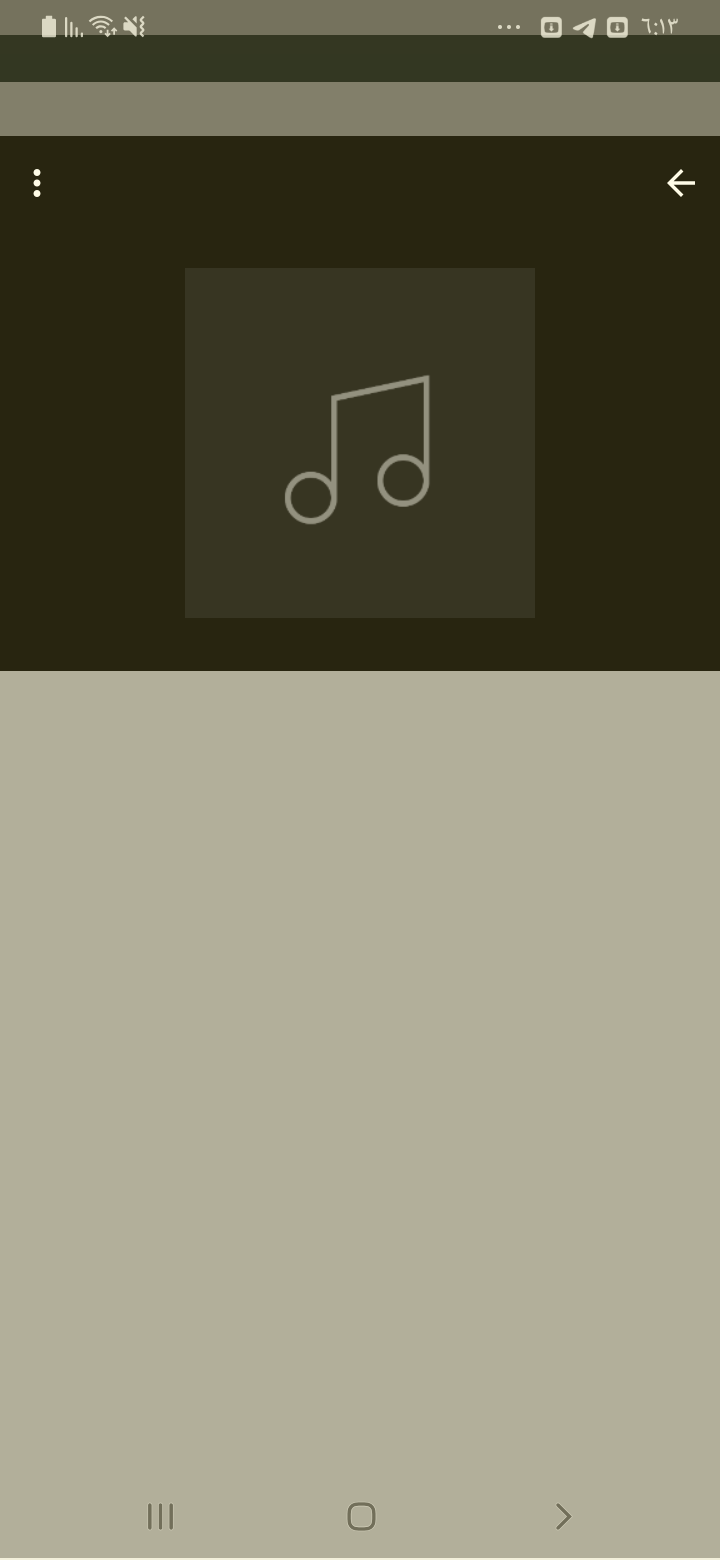
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_home.TestHome object at 0x1104d8d60>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Home Page Tests")  @allure.sub\_suite("Recommended tracks clicking")  @allure.title("Recommended tracks clicking")  @allure.description("Recommended tracks clicking and check that it works")  @pytest.mark.Do  @pytest.mark.HomePage  @pytest.mark.Test5  def test\_case\_5(self, setup):  """  checking the Recommended tracks works  """  ap = AuthenticationPage(self.driver)  ap.click\_signin\_button()  lp = LoginPage(self.driver)  lp.do\_the\_login(Constants.correct\_credentials["email"], Constants.correct\_credentials["password"])  hp = HomePage(self.driver)  time.sleep(10)  hp.click\_recommended\_tracks()  if Helper.element\_exists\_by\_id(self.driver, HomePage.about\_button\_id):  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Home page test failed") > assert False E assert False  Mobile\_Testing/Tests/test\_home.py:198: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# Sign up correctly - failed

Signing up with correct format

## Details

AssertionError: assert False

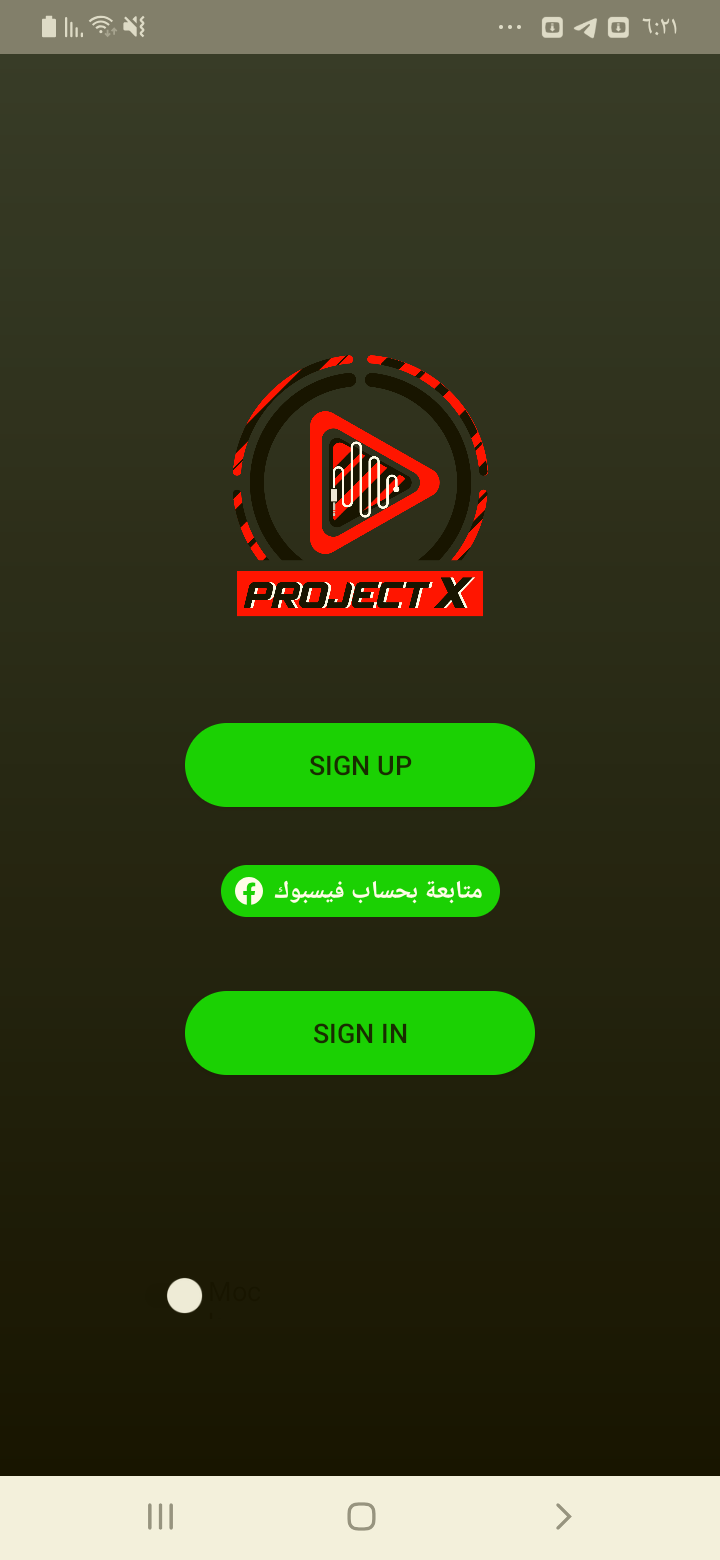
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_signup.TestSignup object at 0x10fc98bb0>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Signup Tests")  @allure.sub\_suite("Sign up correctly")  @allure.title("Sign up correctly")  @allure.description("Signing up with correct format ")  @pytest.mark.Do  @pytest.mark.Signup  @pytest.mark.Test1  def test\_case\_1(self, setup):  """  Signs up correctly  """  ap = AuthenticationPage(self.driver)  ap.click\_signup\_button()  sp = SignupPage(self.driver)  sp.do\_the\_signup(self.name, self.email, self.password, "22", "M")  if Helper.element\_exists\_by\_id(self.driver, HomePage.logout\_button\_id):  Helper.report\_allure(self.driver, "Sign up test Passed")  assert True  else:  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Sign up test Failed") > assert False E assert False  Mobile\_Testing/Tests/test\_signup.py:103: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# Signs up with 999 age - failed

Signs up with 999 age

## Details

AssertionError: assert False

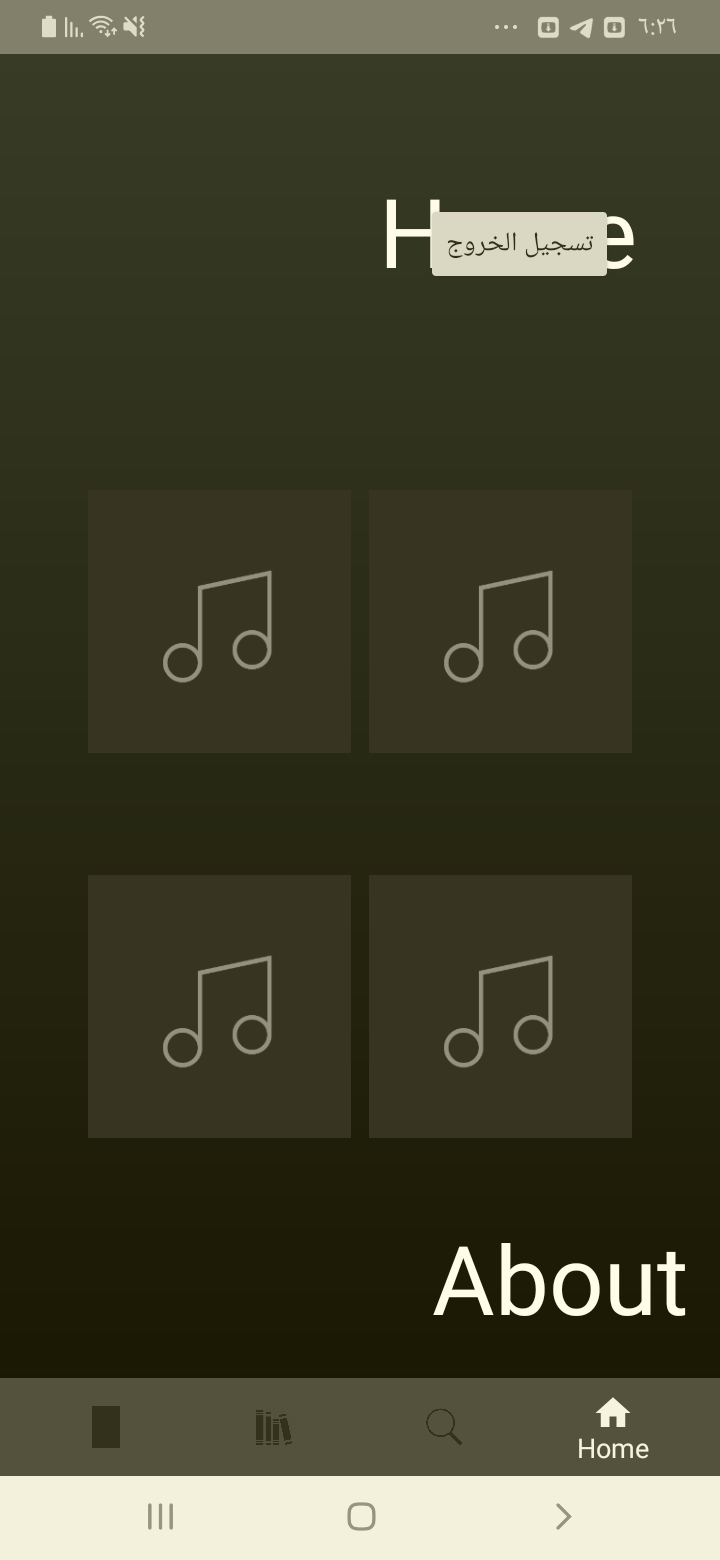
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_signup.TestSignup object at 0x10fd815b0>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Signup Tests")  @allure.sub\_suite("Sign up with 999 age ")  @allure.title("Signs up with 999 age ")  @allure.description("Signs up with 999 age")  @pytest.mark.Do  @pytest.mark.Signup  @pytest.mark.Test12  def test\_case\_12(self, setup):  """  Signs up with 999 age  """  ap = AuthenticationPage(self.driver)  ap.click\_signup\_button()  sp = SignupPage(self.driver)  sp.do\_the\_signup(self.name, self.email, self.password, "999", "M")  if Helper.element\_exists\_by\_id(self.driver, HomePage.logout\_button\_id):  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Sign up test Failed") > assert False E assert False  Mobile\_Testing/Tests/test\_signup.py:376: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# Signs up with wrong name format - failed

Signs up with wrong name format

## Details

AssertionError: assert False

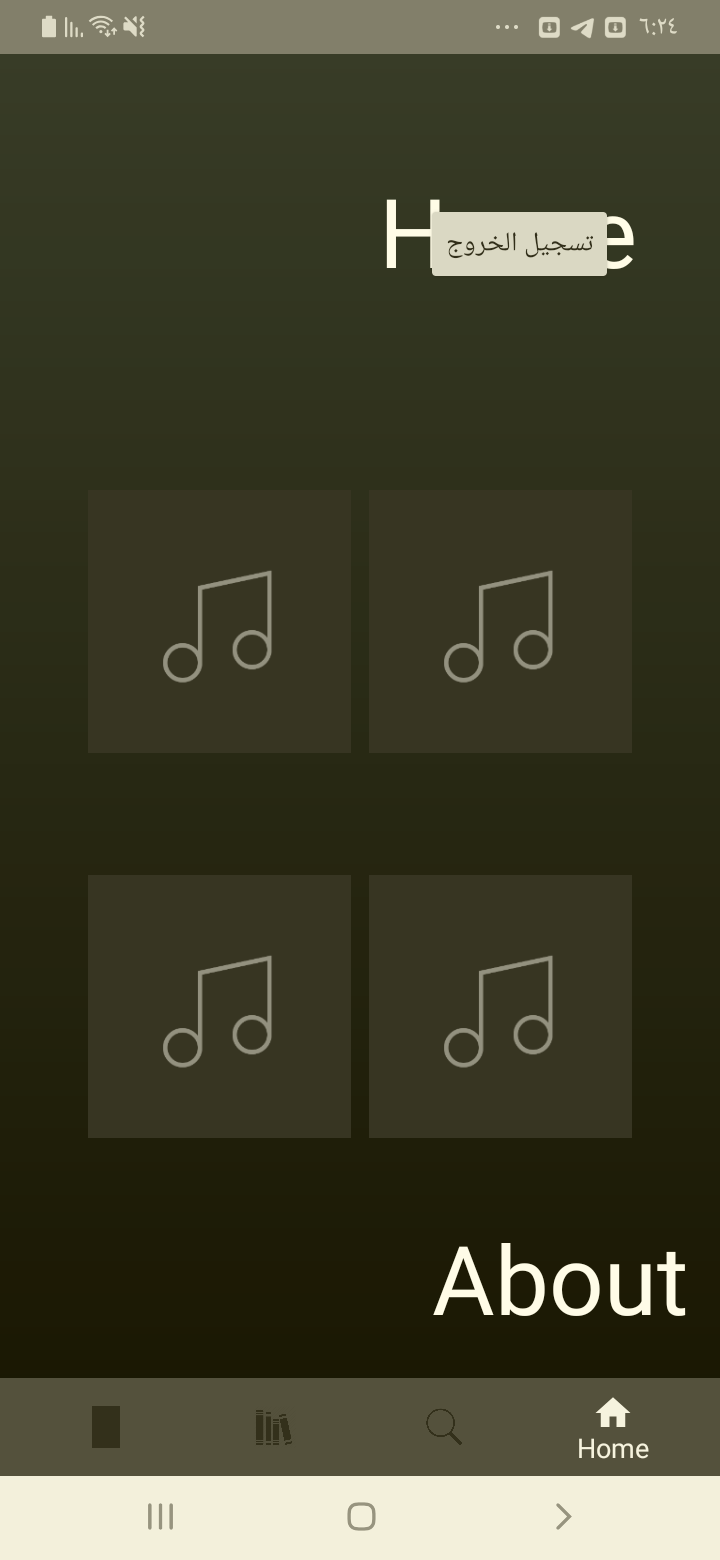
|  |
| --- |
| self = <Mobile\_Testing.Tests.test\_signup.TestSignup object at 0x10fd81af0>, setup = None   @allure.severity(allure.severity\_level.BLOCKER)  @allure.story("Signup Tests")  @allure.sub\_suite("Sign up with wrong name format")  @allure.title("Signs up with wrong name format")  @allure.description("Signs up with wrong name format")  @pytest.mark.Do  @pytest.mark.Signup  @pytest.mark.Test8  def test\_case\_8(self, setup):  """  Signs up with wrong name format  """  ap = AuthenticationPage(self.driver)  ap.click\_signup\_button()  sp = SignupPage(self.driver)  sp.do\_the\_signup("@!@#!@", self.email, self.password, "21", "M")  if Helper.element\_exists\_by\_id(self.driver, HomePage.logout\_button\_id):  print(self.driver.current\_activity)  Helper.report\_allure(self.driver, "Sign up test Failed") > assert False E assert False  Mobile\_Testing/Tests/test\_signup.py:276: AssertionError |

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Failed



[Attachment] stdout

## Test Teardown

[Fixture] setup::0

# About clicking - passed

About clicking and check that it works

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test passed



## Test Teardown

[Fixture] setup::0

# Blacklist button checking - passed

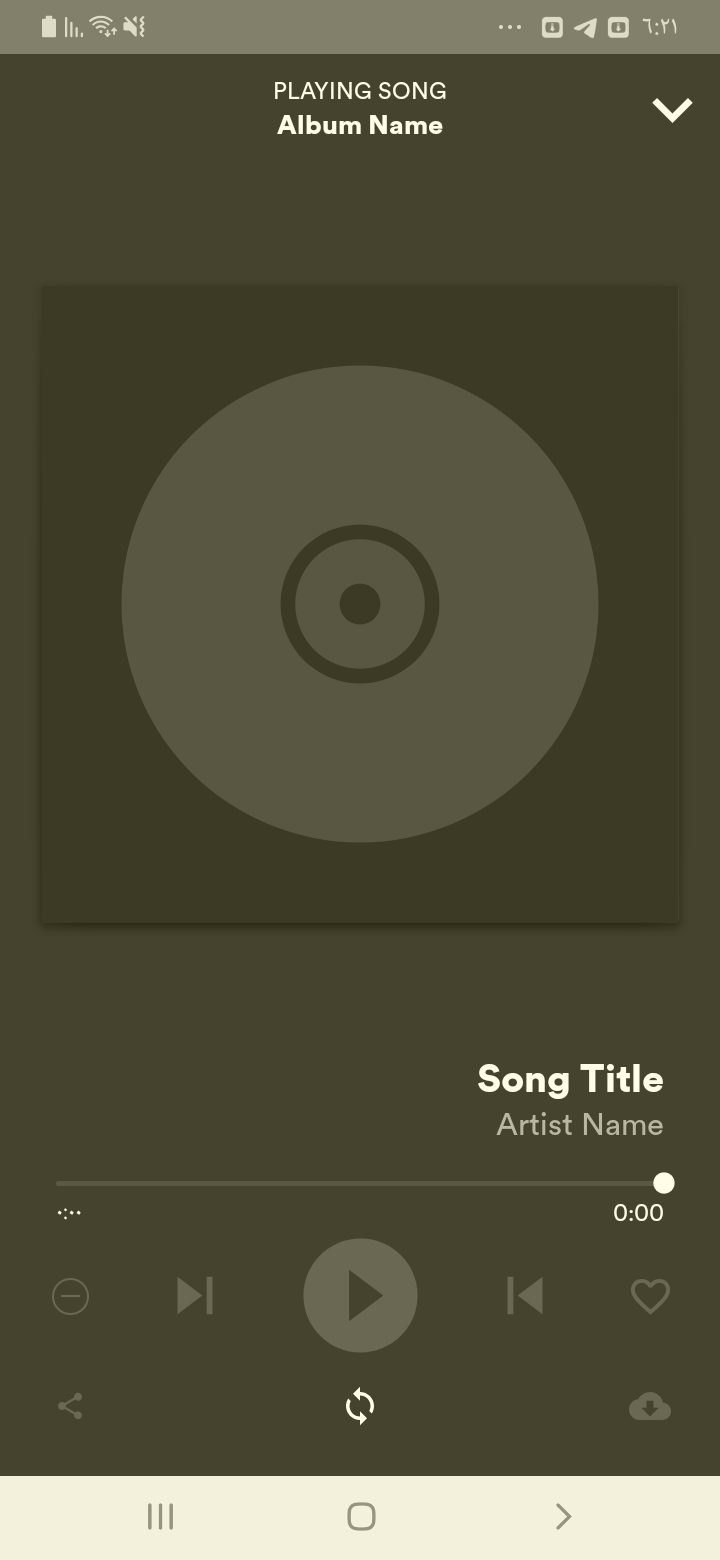
Checking that the Blacklist button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Download button checking - passed

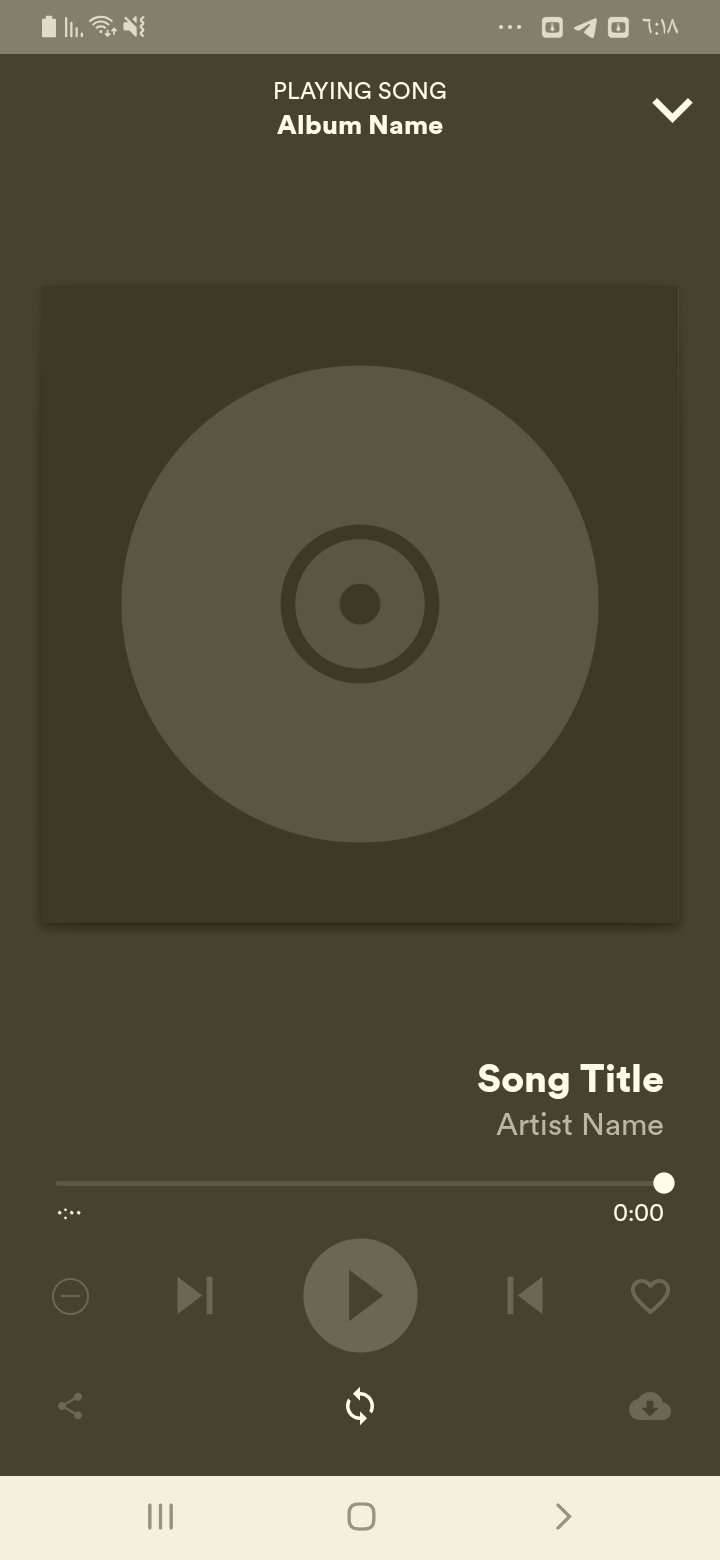
Checking that the Download button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Facebook Button Checking - passed

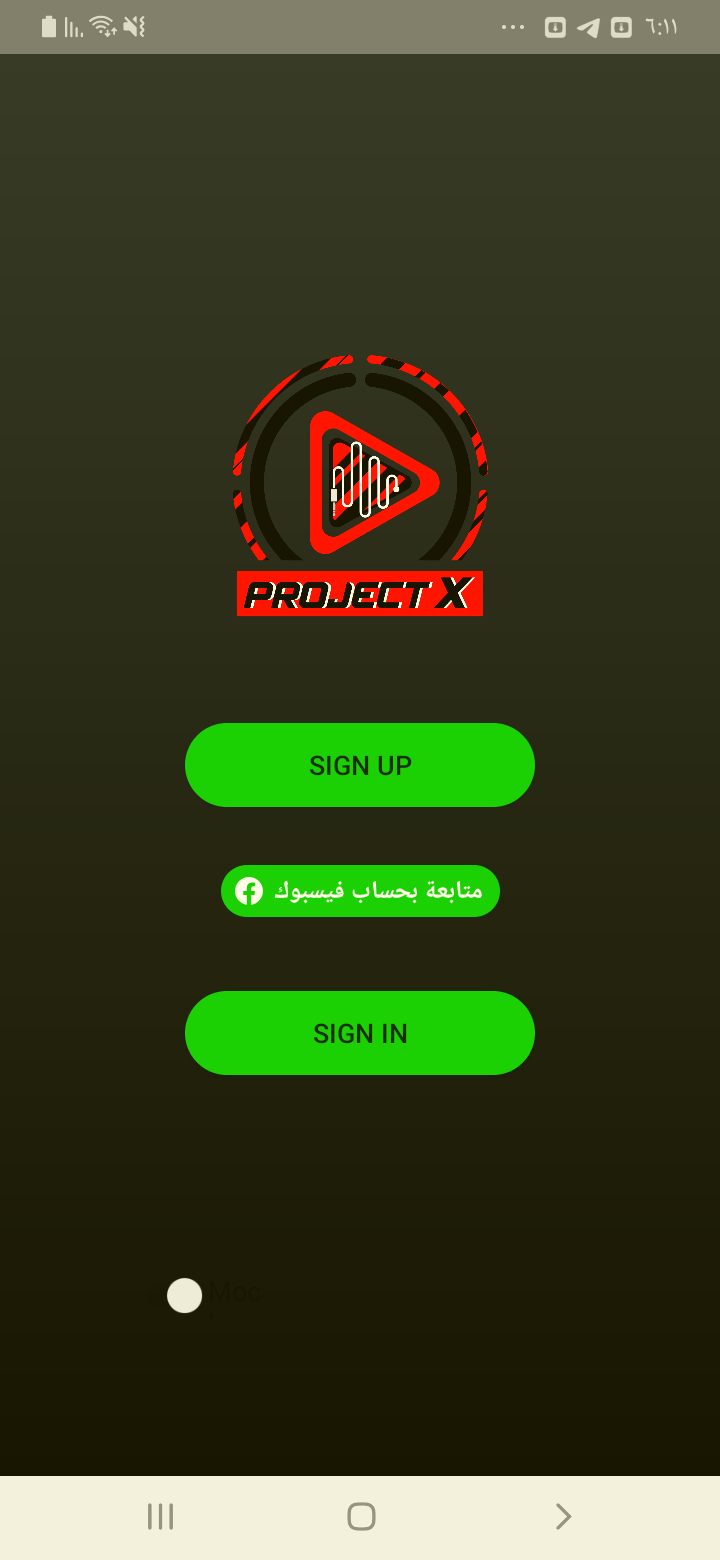
Checking the Facebook button is available

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Facebook Page available



## Test Teardown

[Fixture] setup::0

# Home clicking - passed

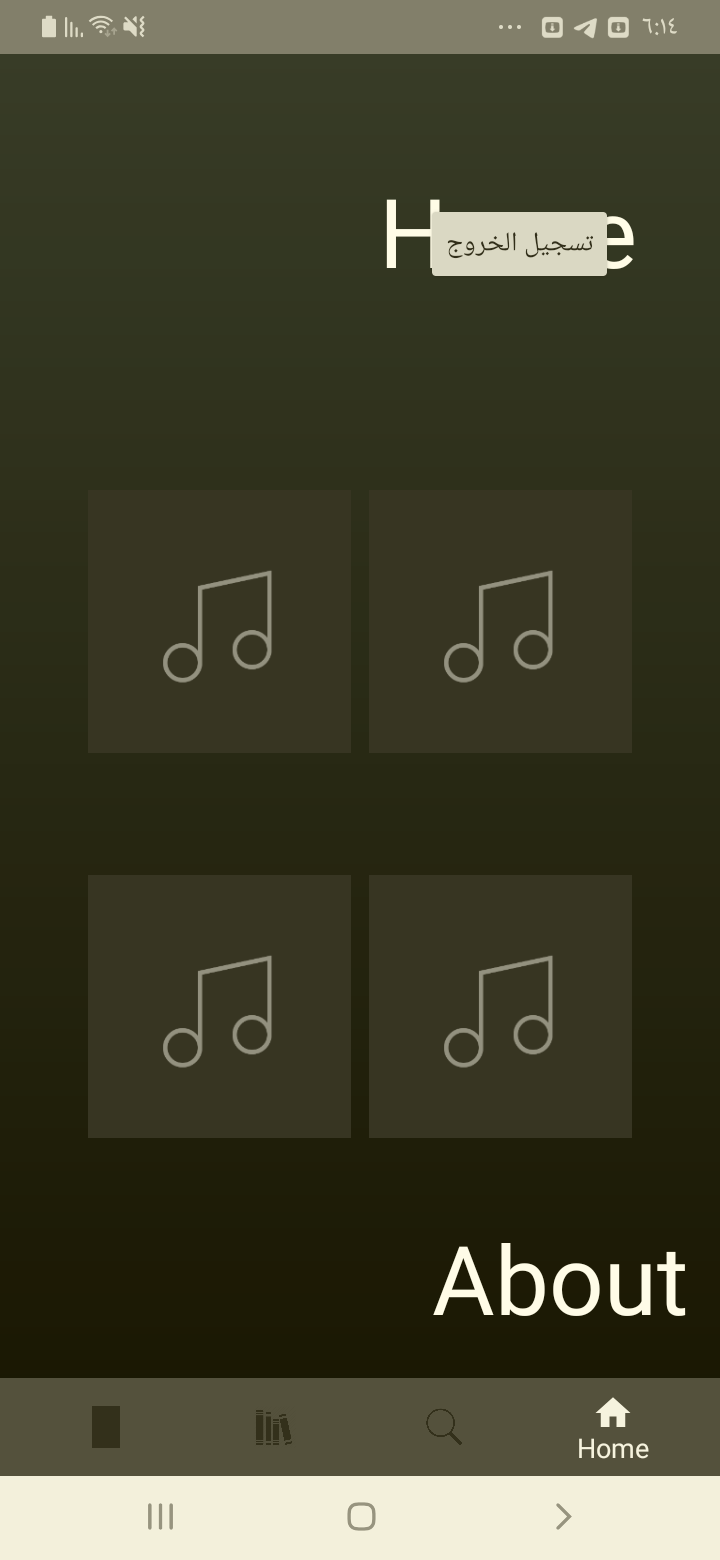
Home clicking and check that it works

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test passed



## Test Teardown

[Fixture] setup::0

# Login clicking - passed

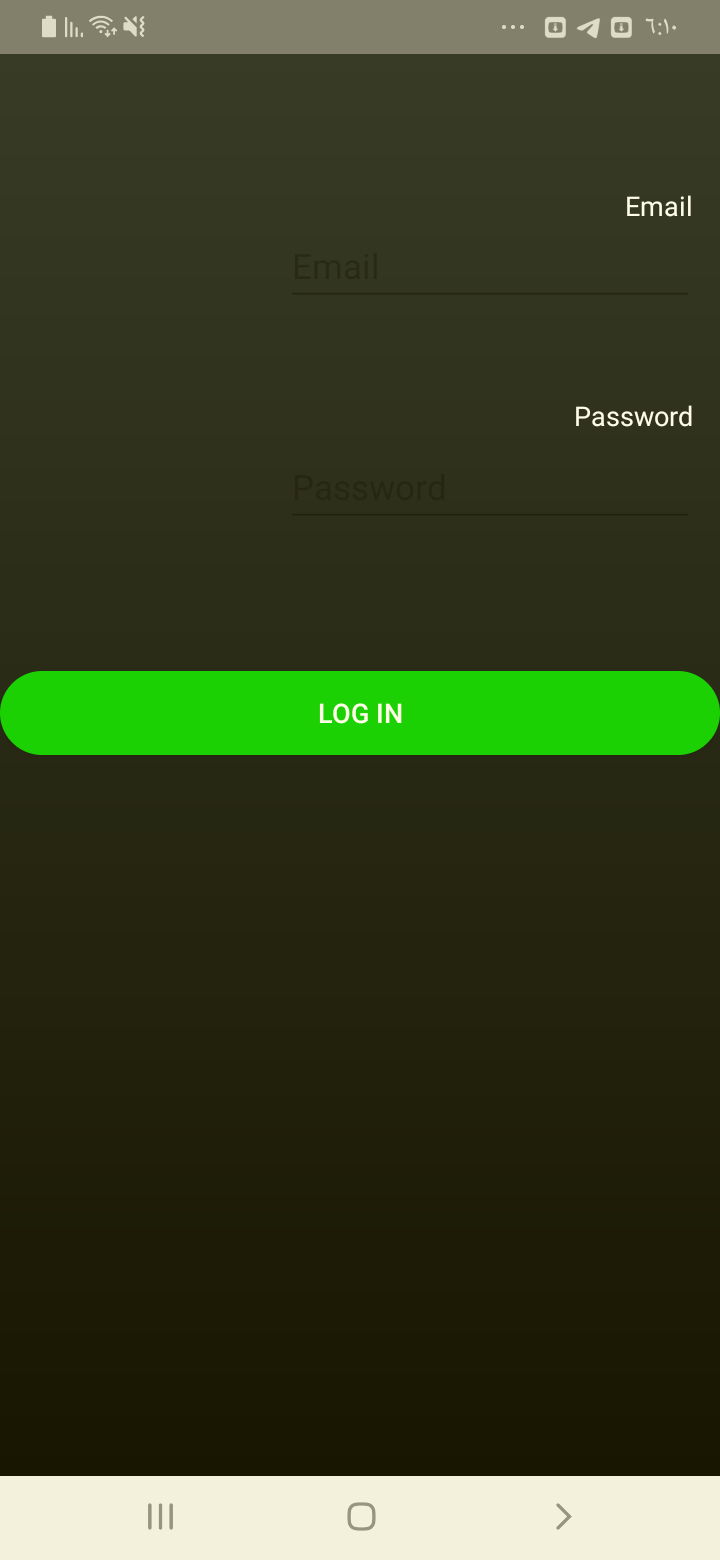
Clicking the Login button

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login Page entered



## Test Teardown

[Fixture] setup::0

# Login correctly - passed

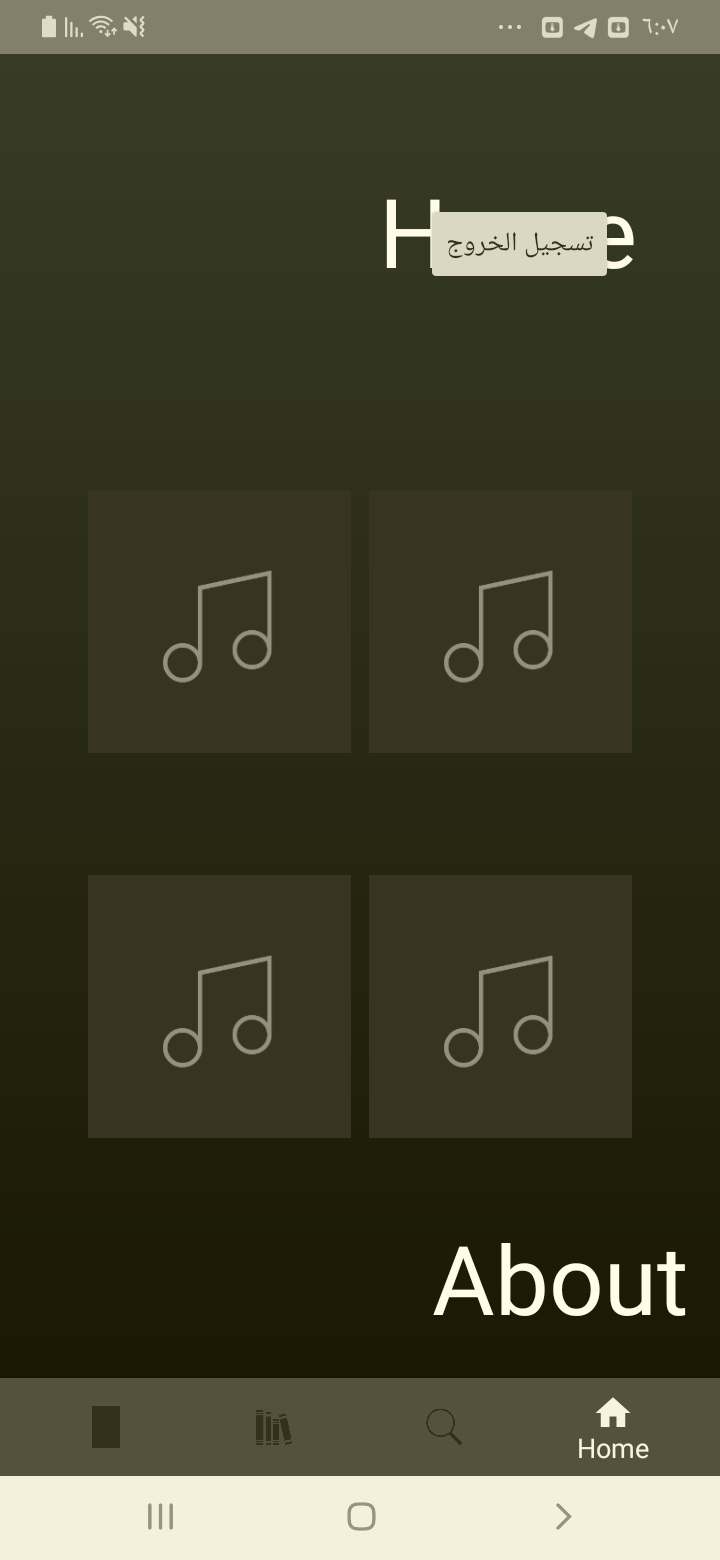
Logging in with correct format with abdallah@gmail.com, 1234567

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Login with no password & email - passed

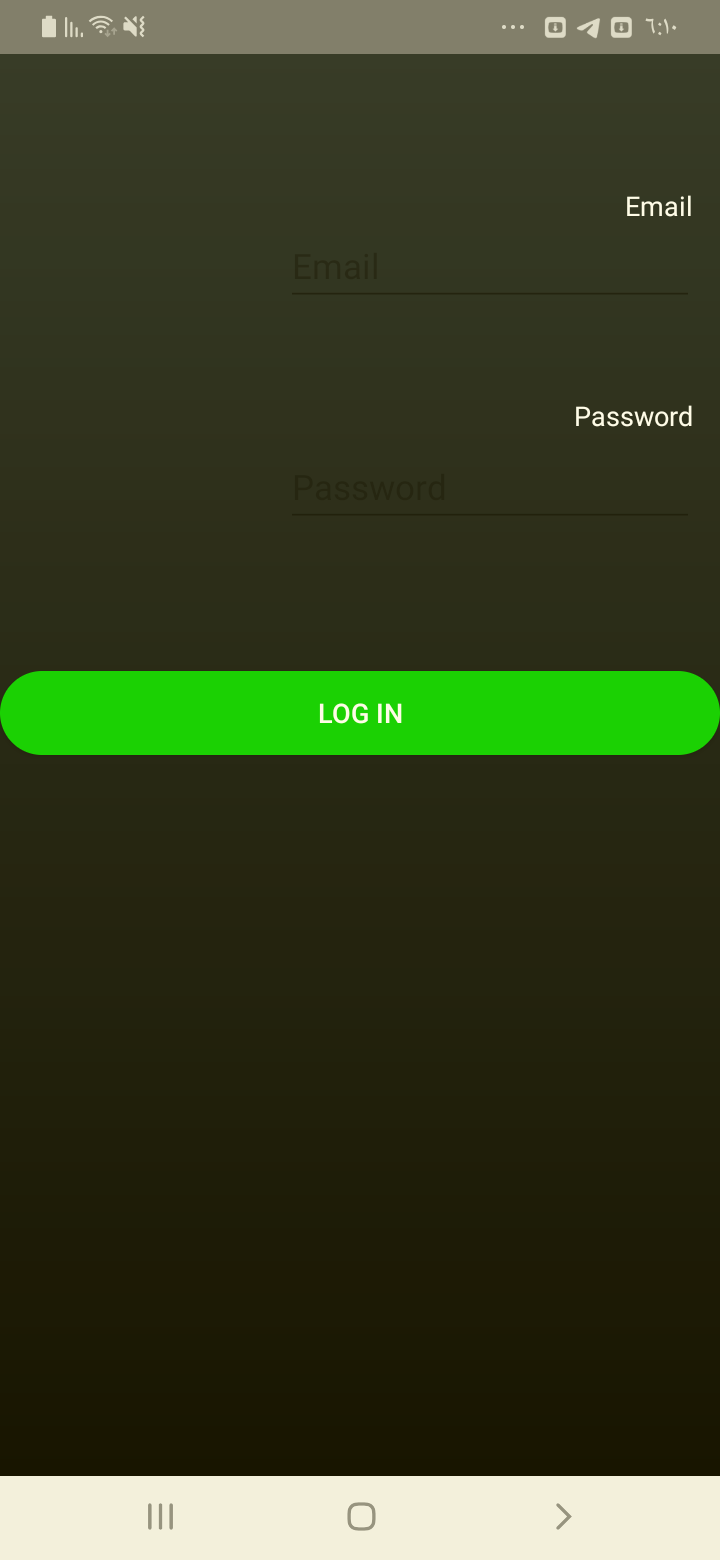
Logging in with no password & email with ,

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Login with no email - passed

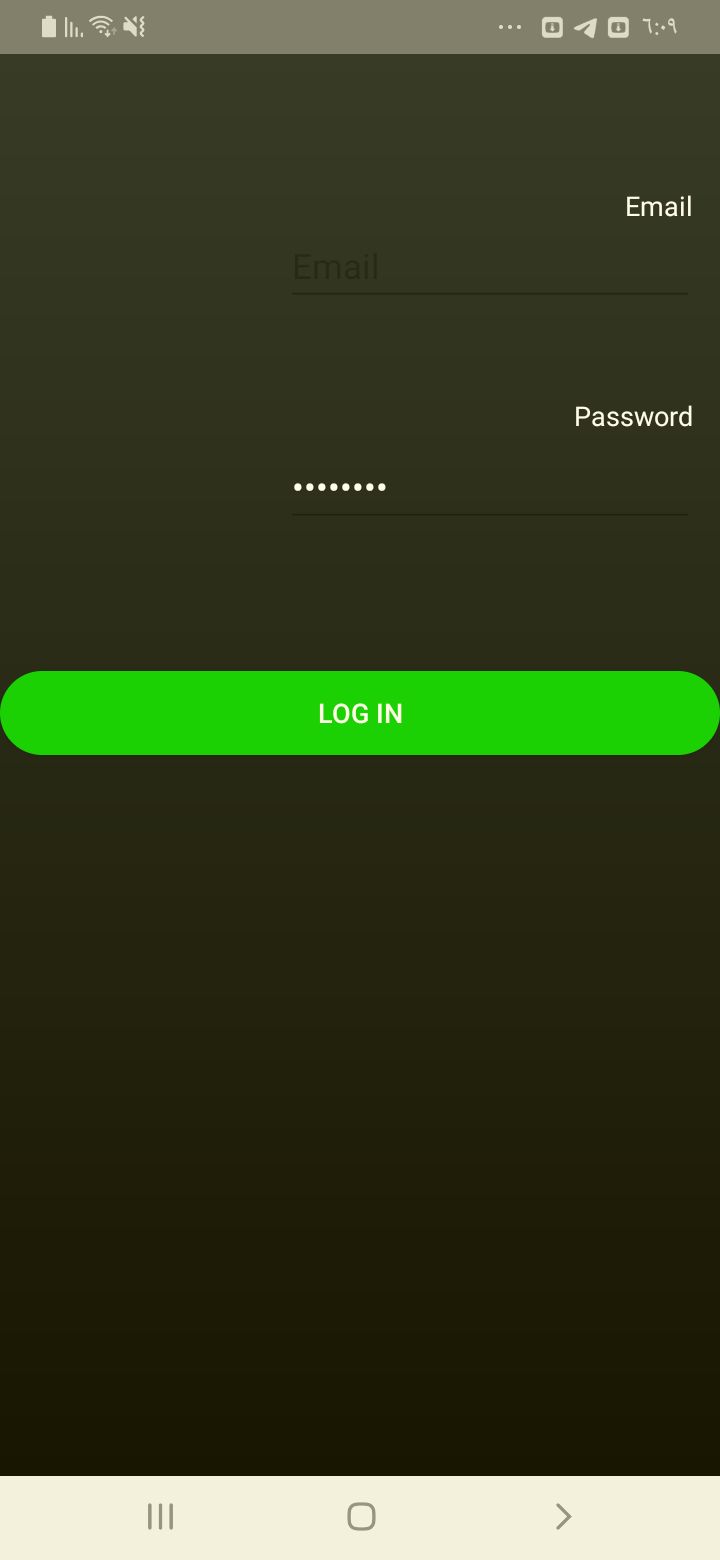
Logging in with no email with , Kimo2010

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Login with no password - passed

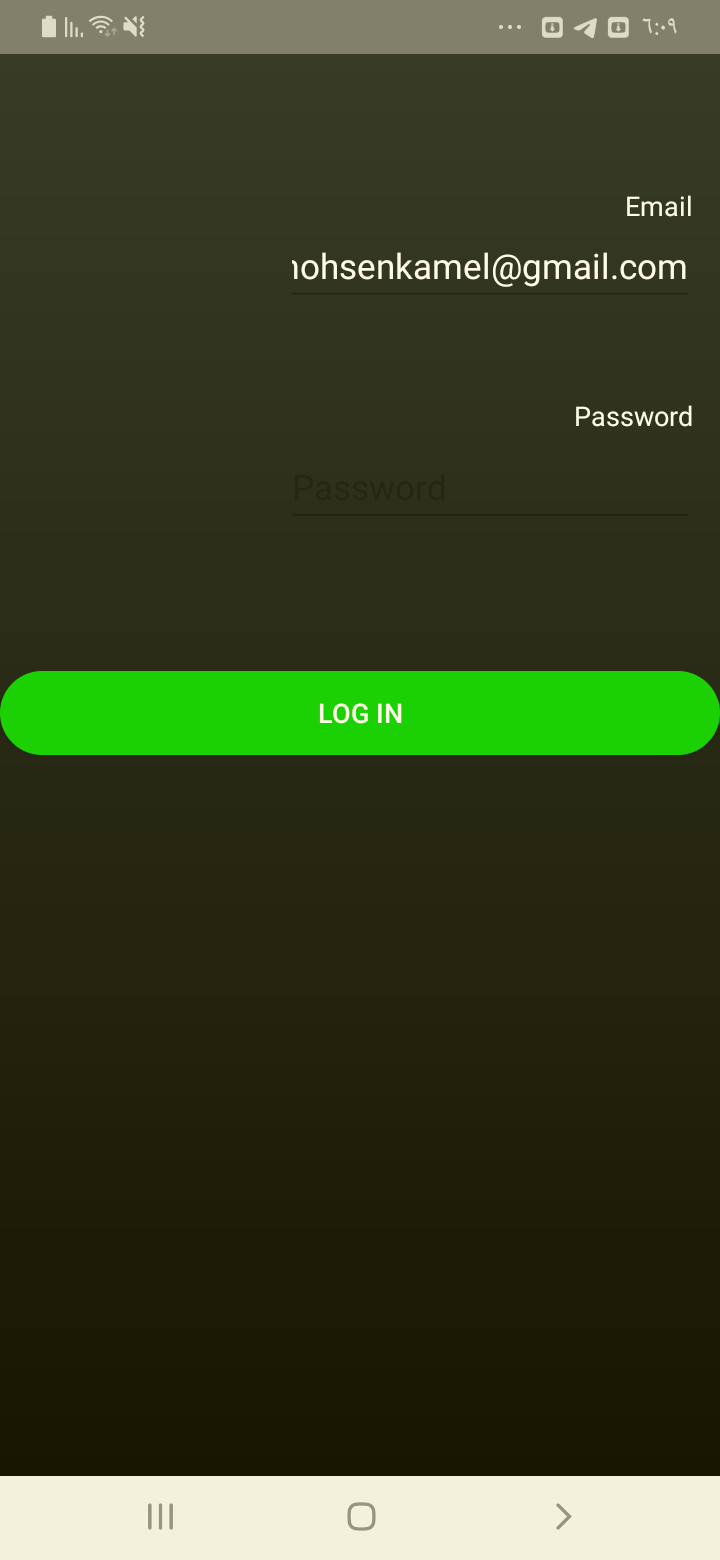
Logging in with no password with kamelmohsenkamel@gmail.com,

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Login with wrong email - passed

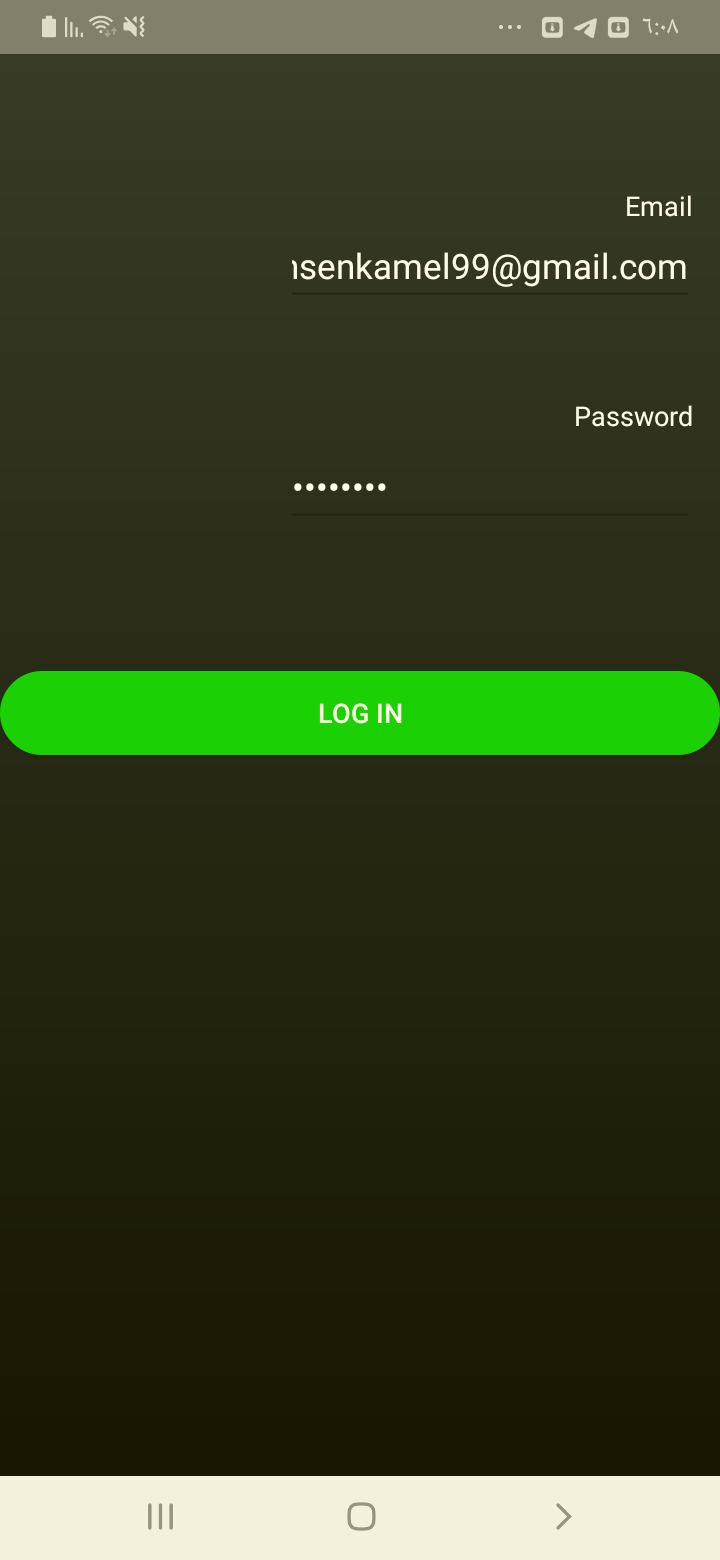
Logging in with wrong email with kamelmohsenkamel99@gmail.com, Kimo2010

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Login with wrong email format - passed

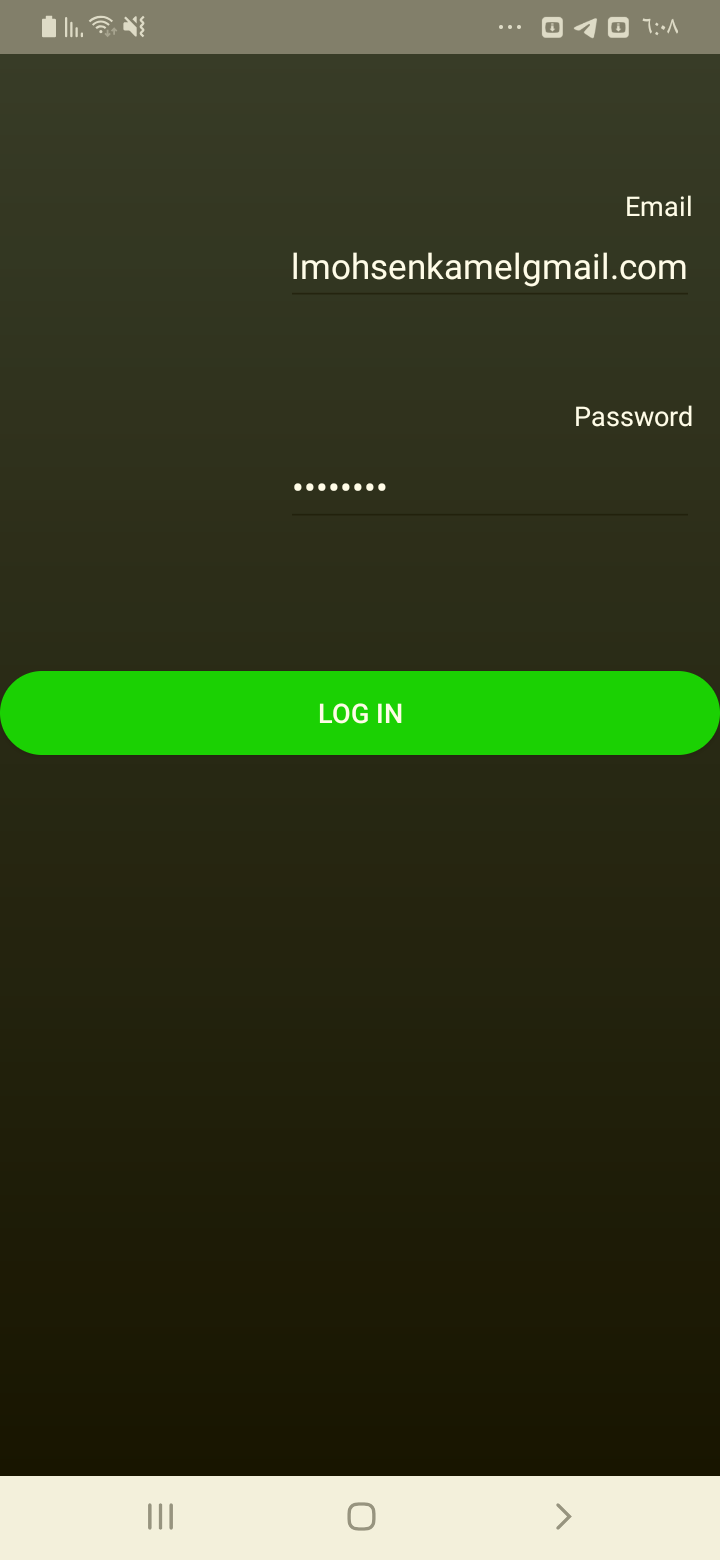
Logging in with wrong email format with kamelmohsenkamelgmail.com, Kimo2010

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Login with wrong password - passed

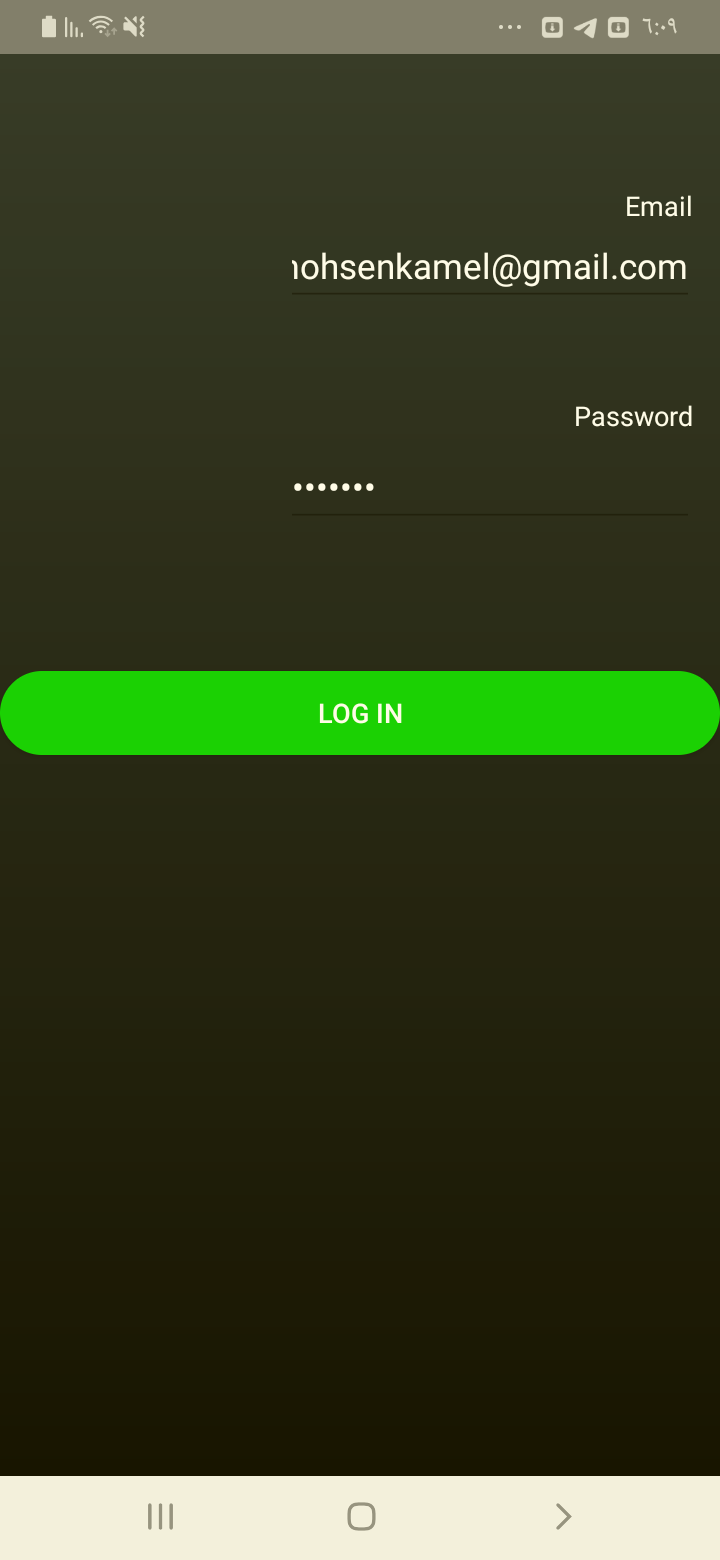
Logging in with wrong password with kamelmohsenkamel@gmail.com, !@#!@#1

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Login test passed



## Test Teardown

[Fixture] setup::0

# Logout clicking - passed

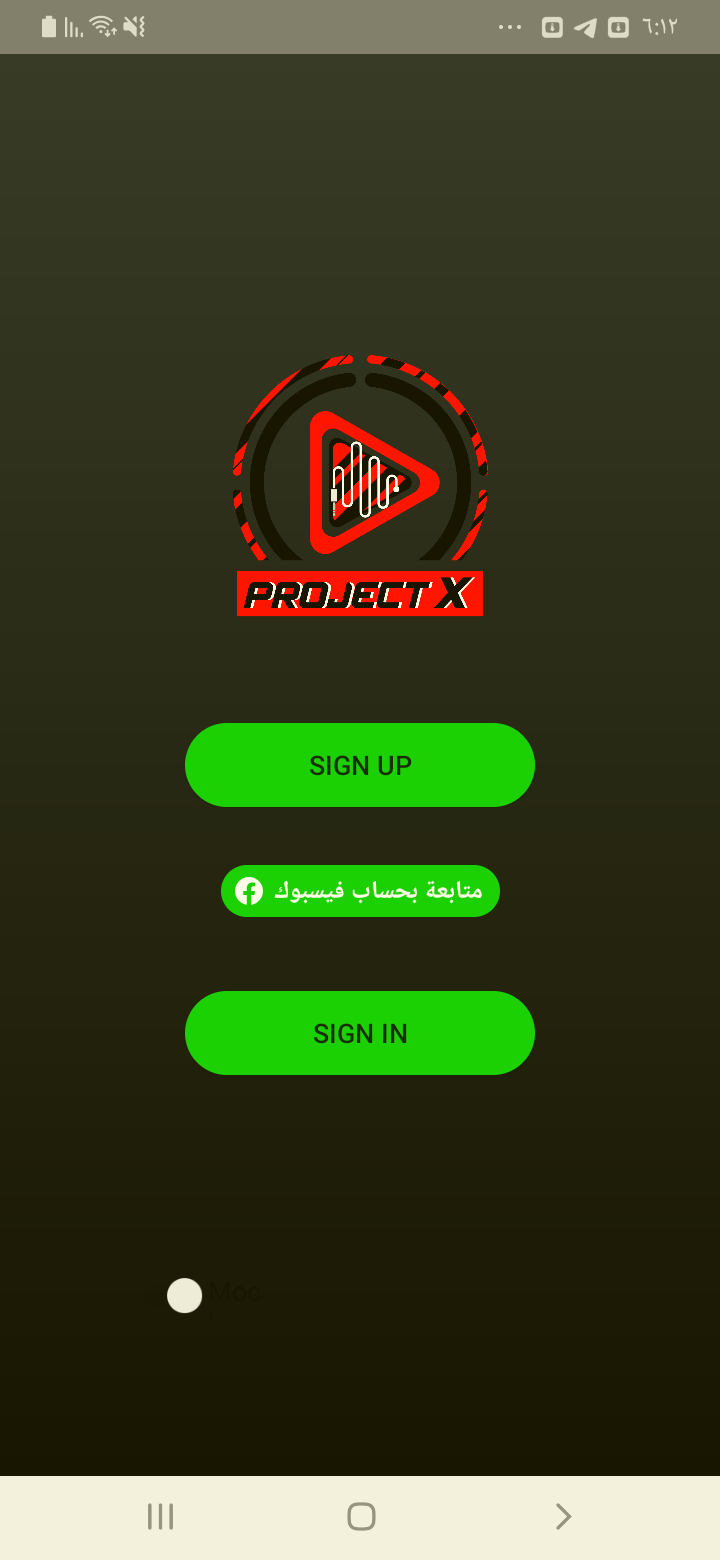
Logout clicking and check that it works

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test passed



## Test Teardown

[Fixture] setup::0

# Love button checking - passed

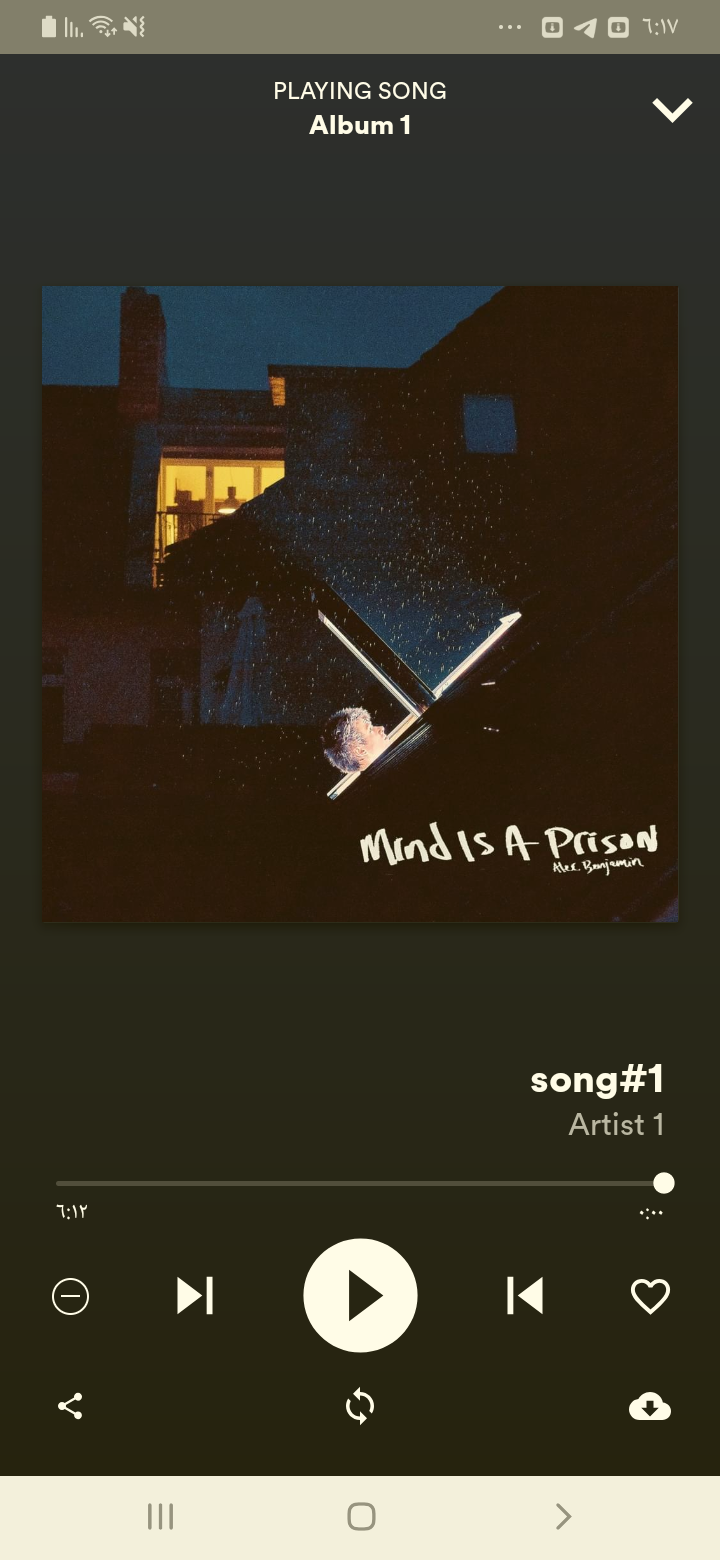
Checking that the Love button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Minimize button checking - passed

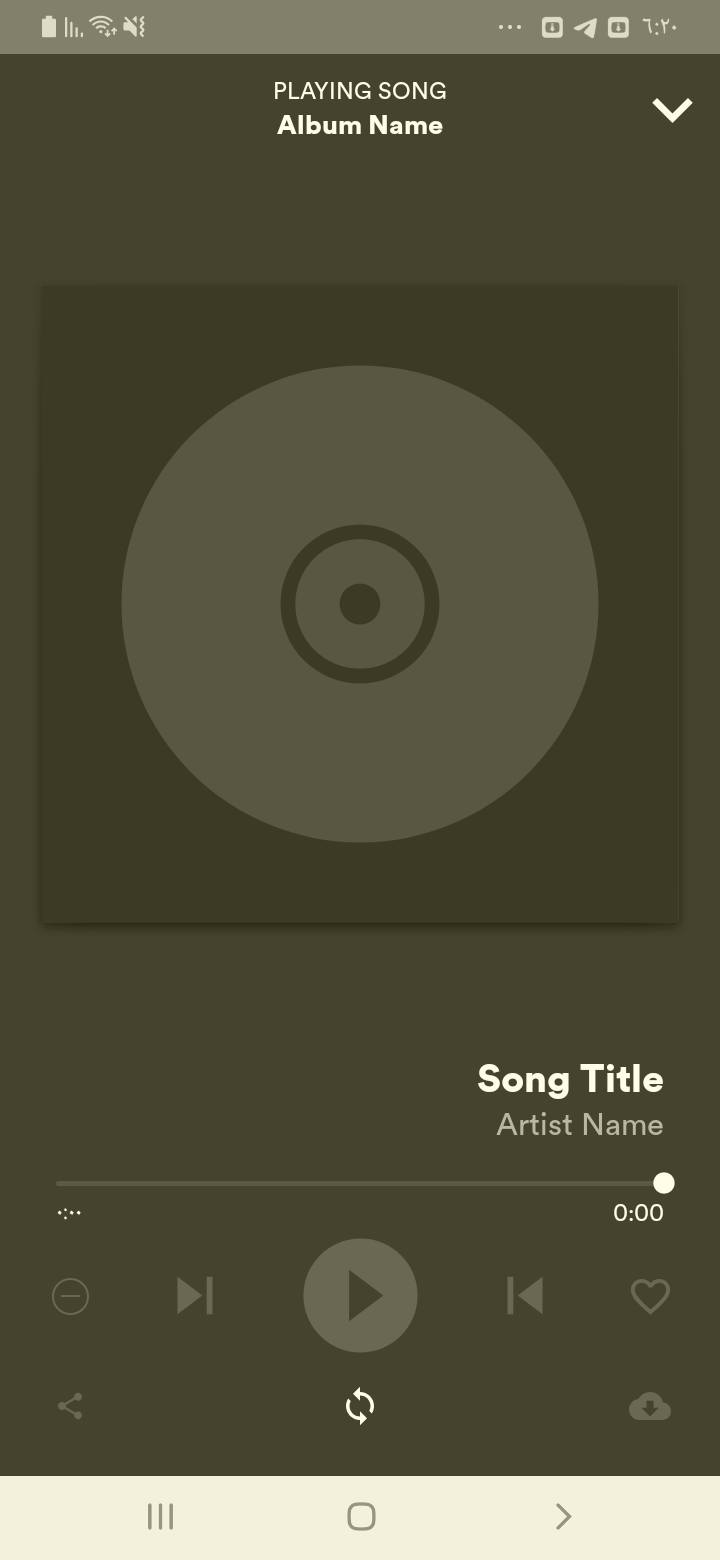
Checking that the Minimize button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Next button checking - passed

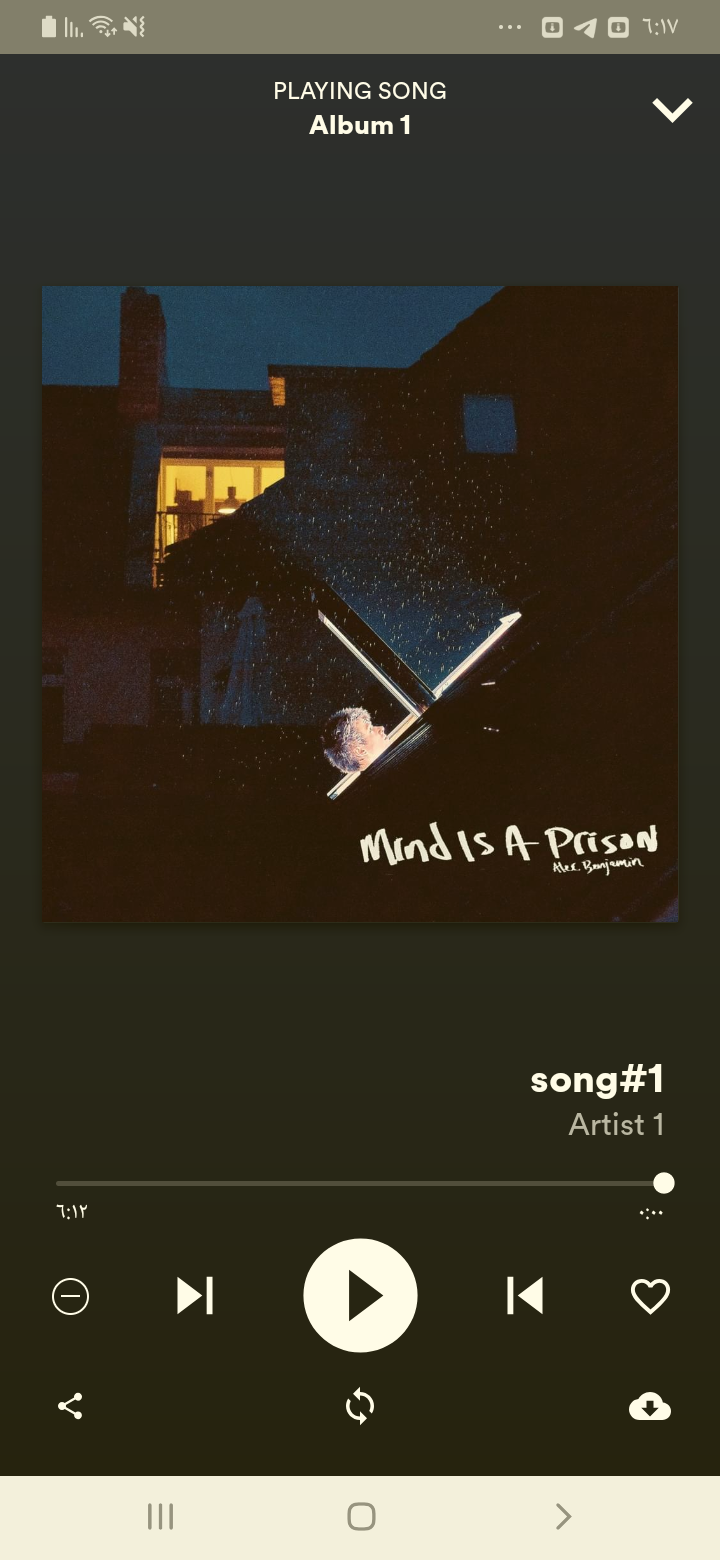
Checking that the Next button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Play button checking - passed

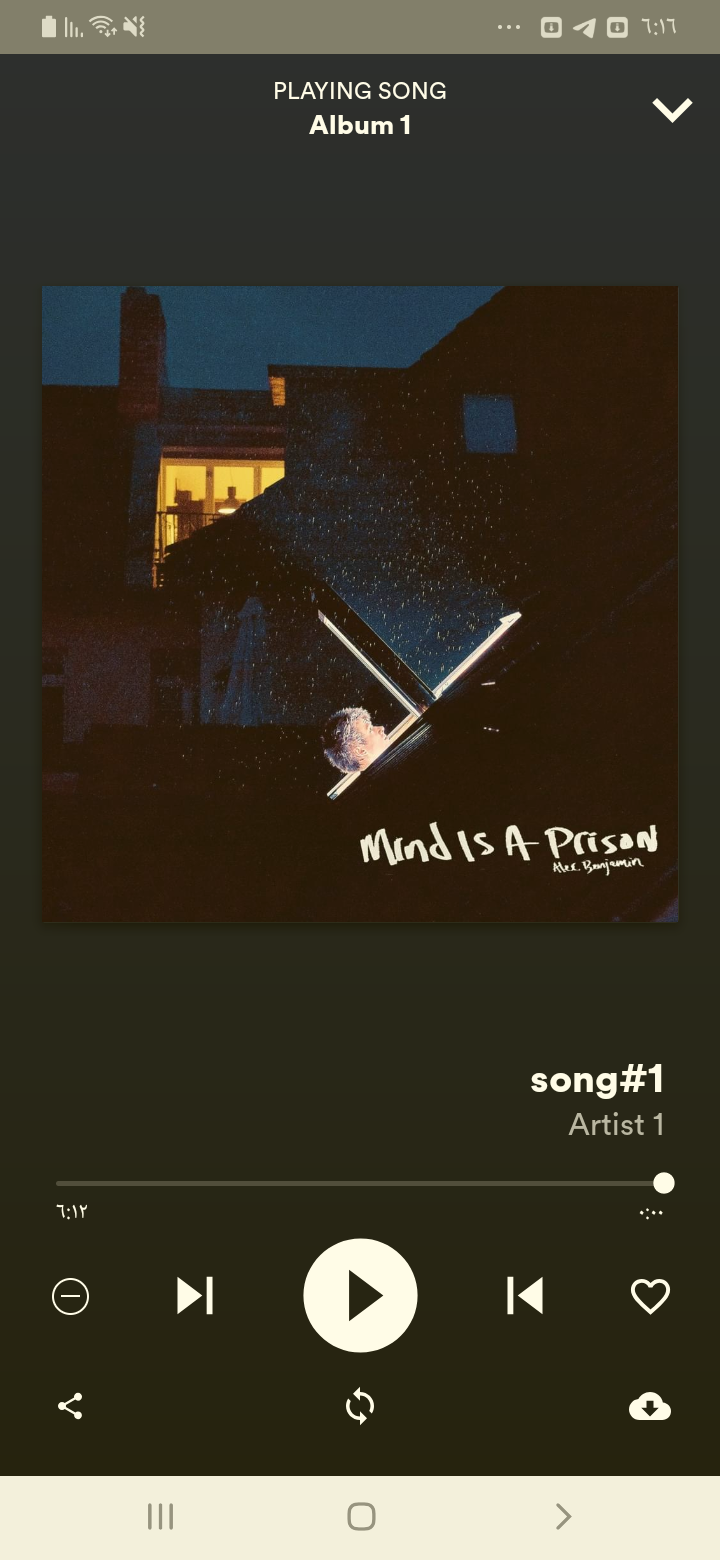
Checking that the Play button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Previous button checking - passed

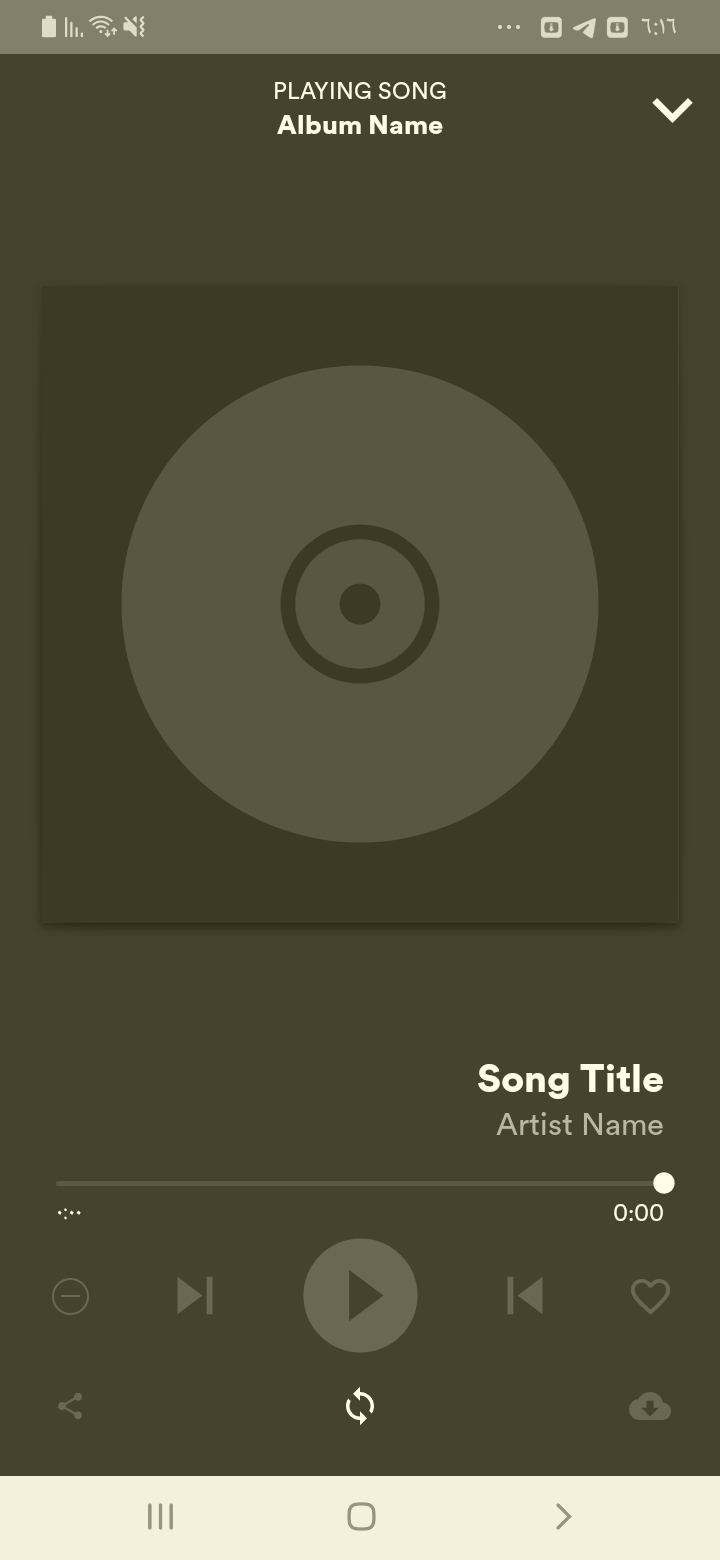
Checking that the Previous button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Repeat button checking - passed

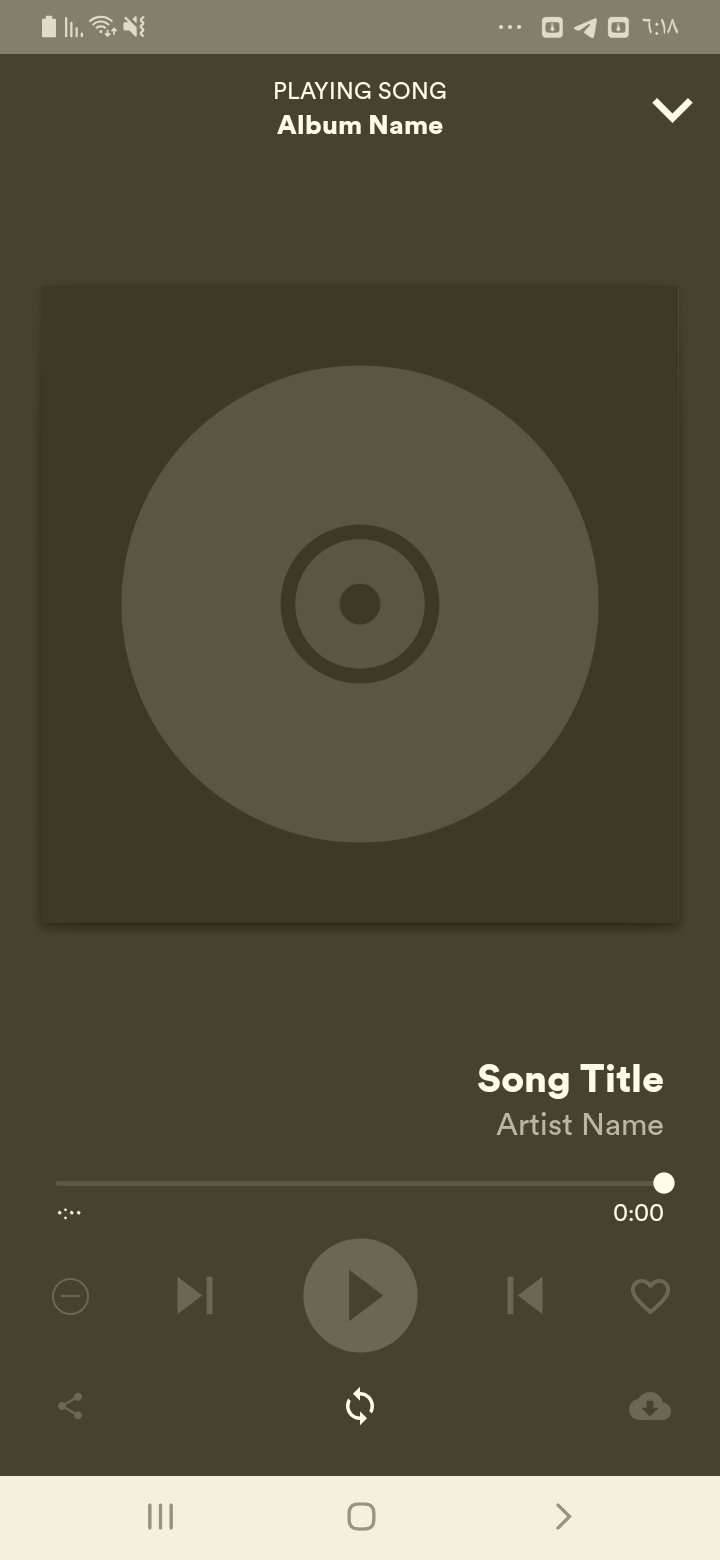
Checking that the repeat button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Search clicking - passed

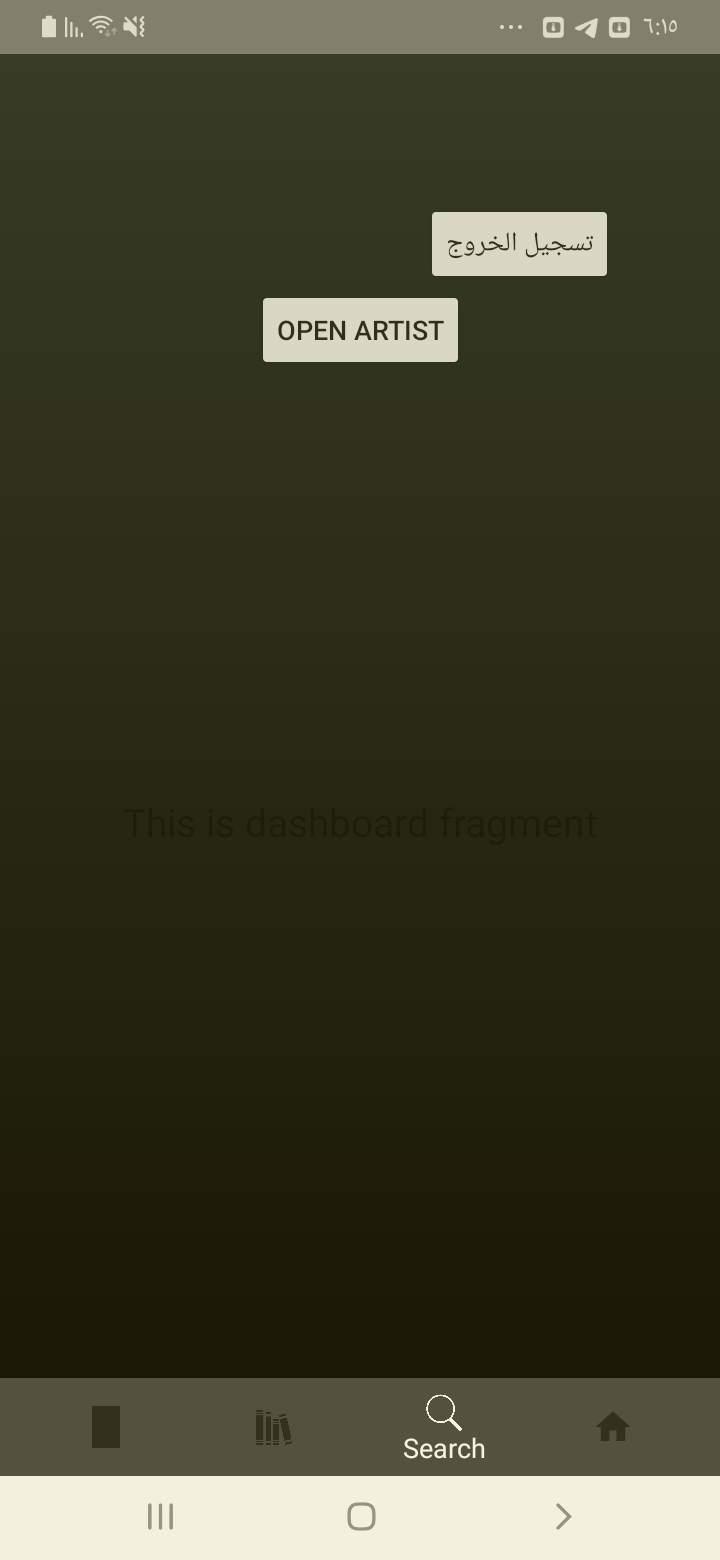
Search clicking and check that it works

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Home page test passed



## Test Teardown

[Fixture] setup::0

# Share button checking - passed

Checking that the Share button is visible

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Player page test passed



## Test Teardown

[Fixture] setup::0

# Sign up clicking - passed

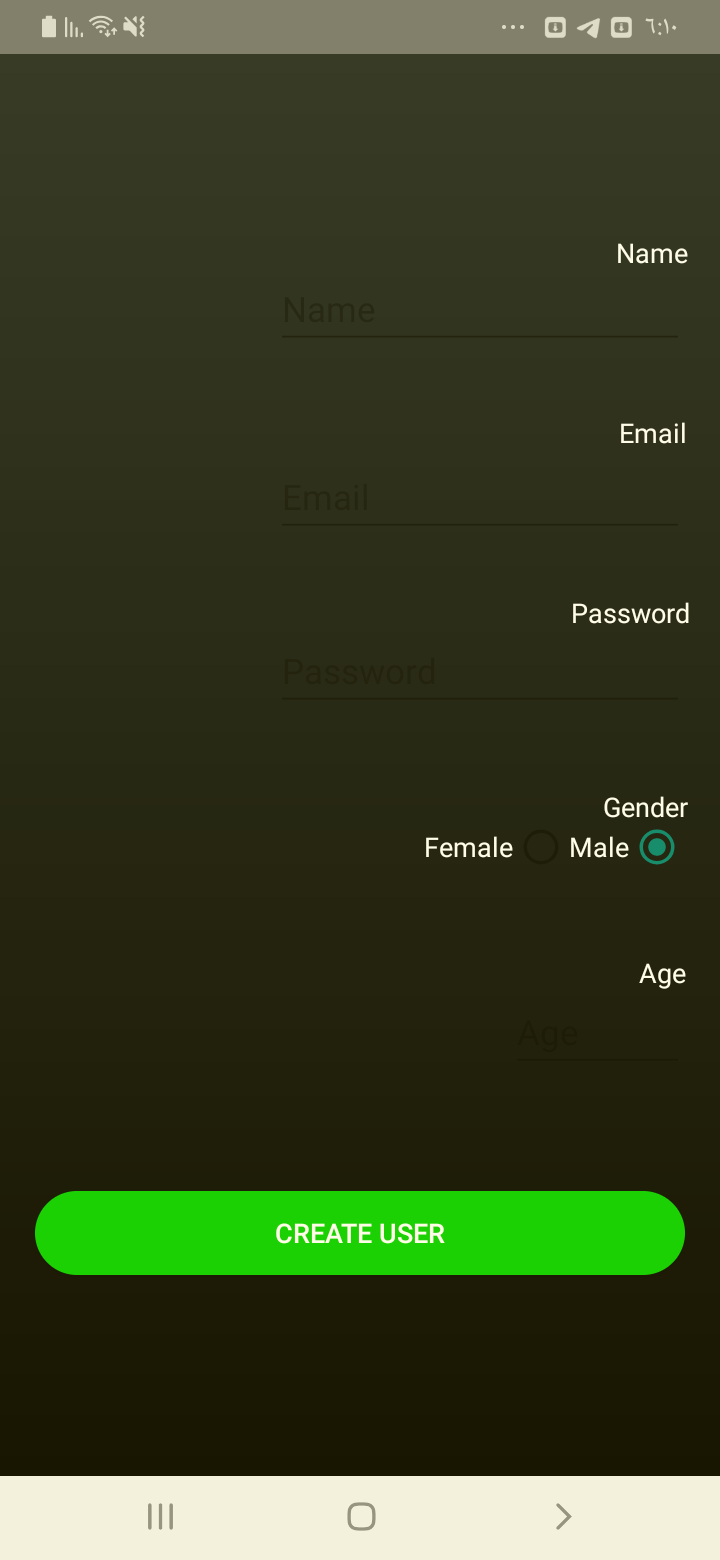
Clicking the sign up button

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up Page entered



## Test Teardown

[Fixture] setup::0

# Signs up with -ve age - passed

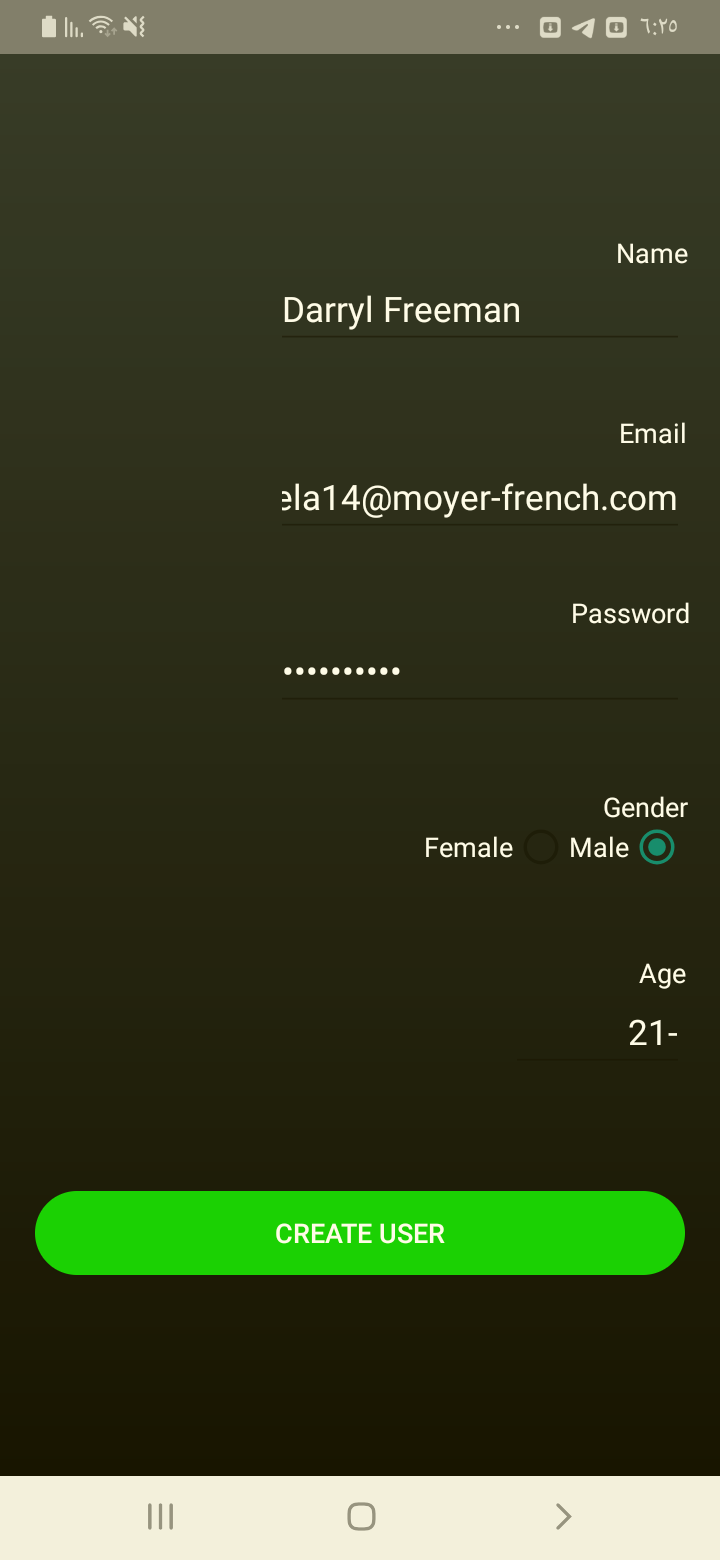
Signs up with -ve age

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with 0 age - passed

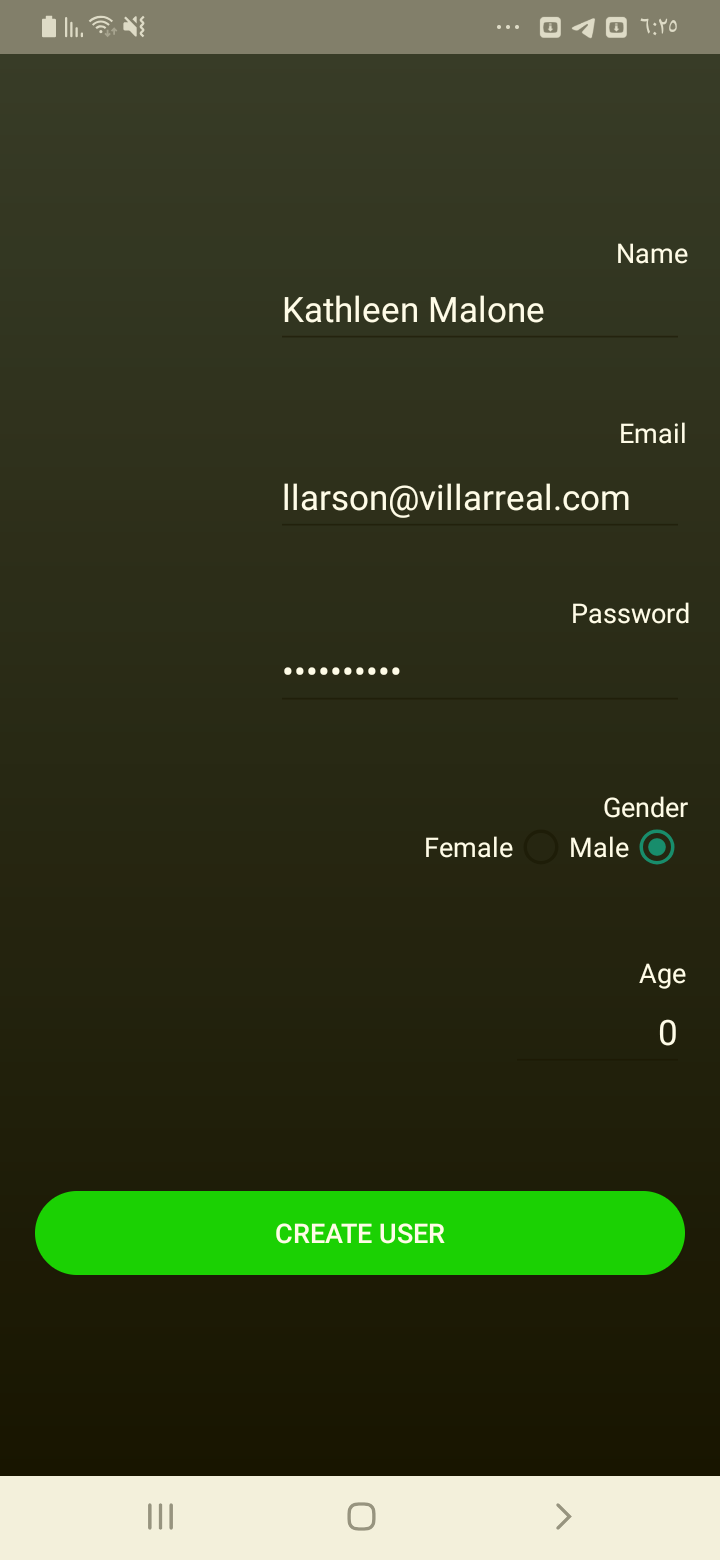
Signs up with 0 age

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with existing email - passed

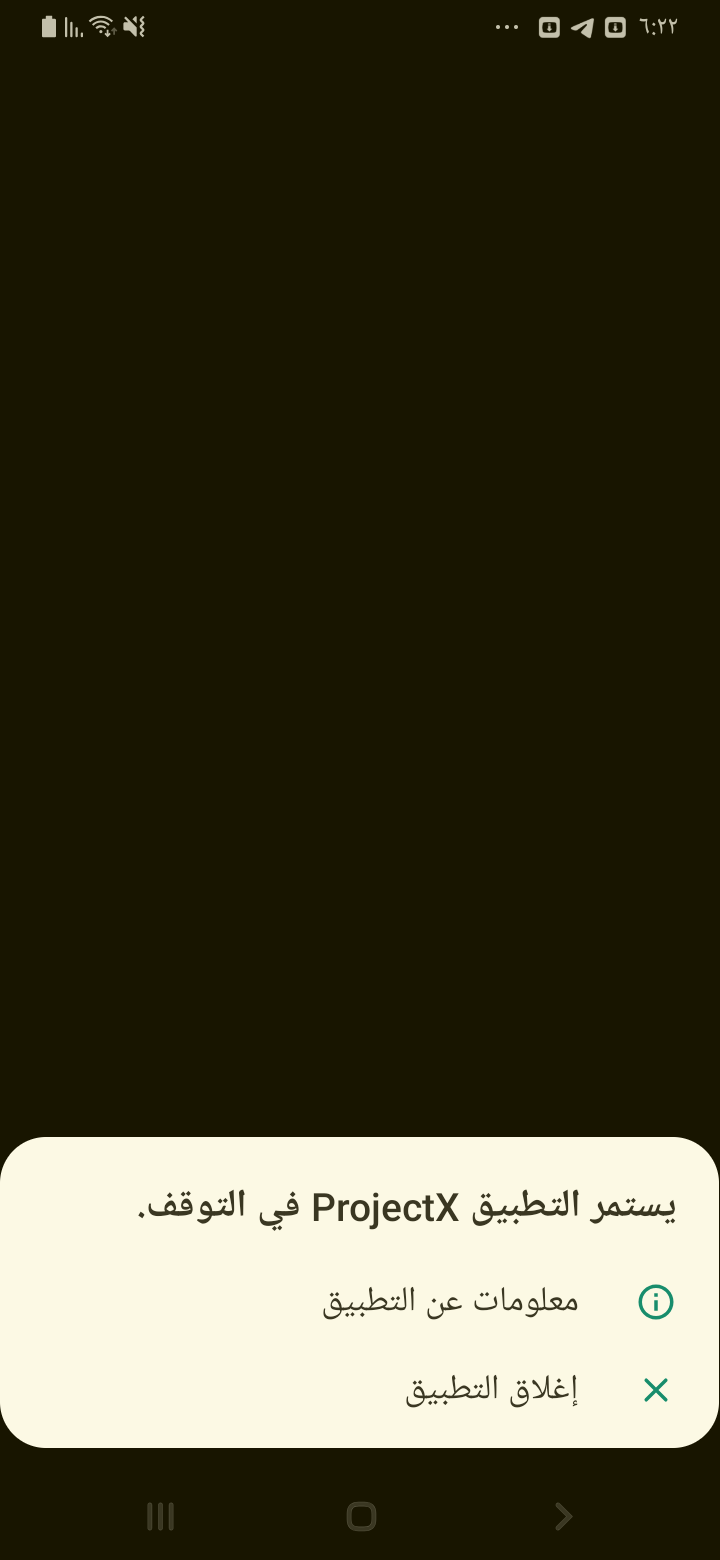
Signs up with existing email

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with female gender - passed

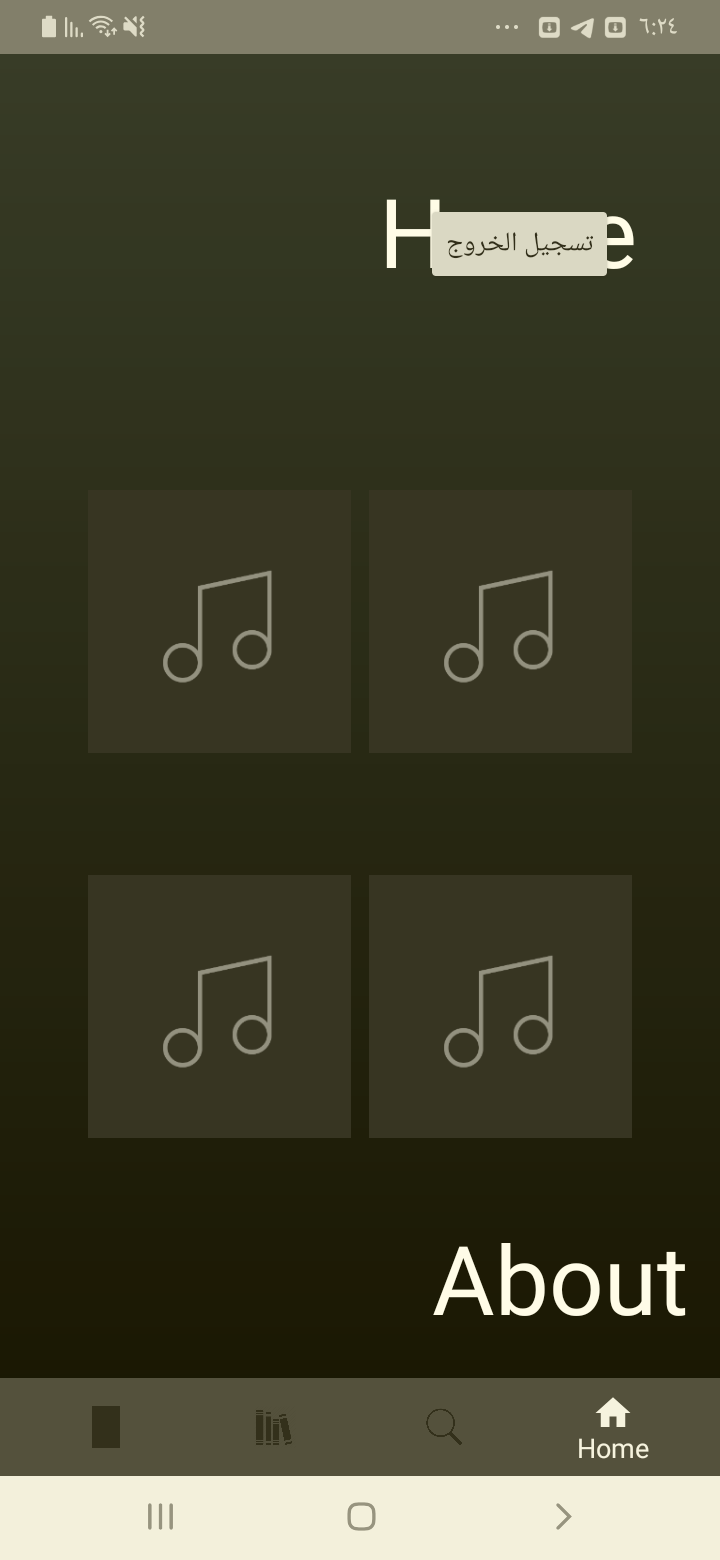
Signs up with female

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with no age - passed

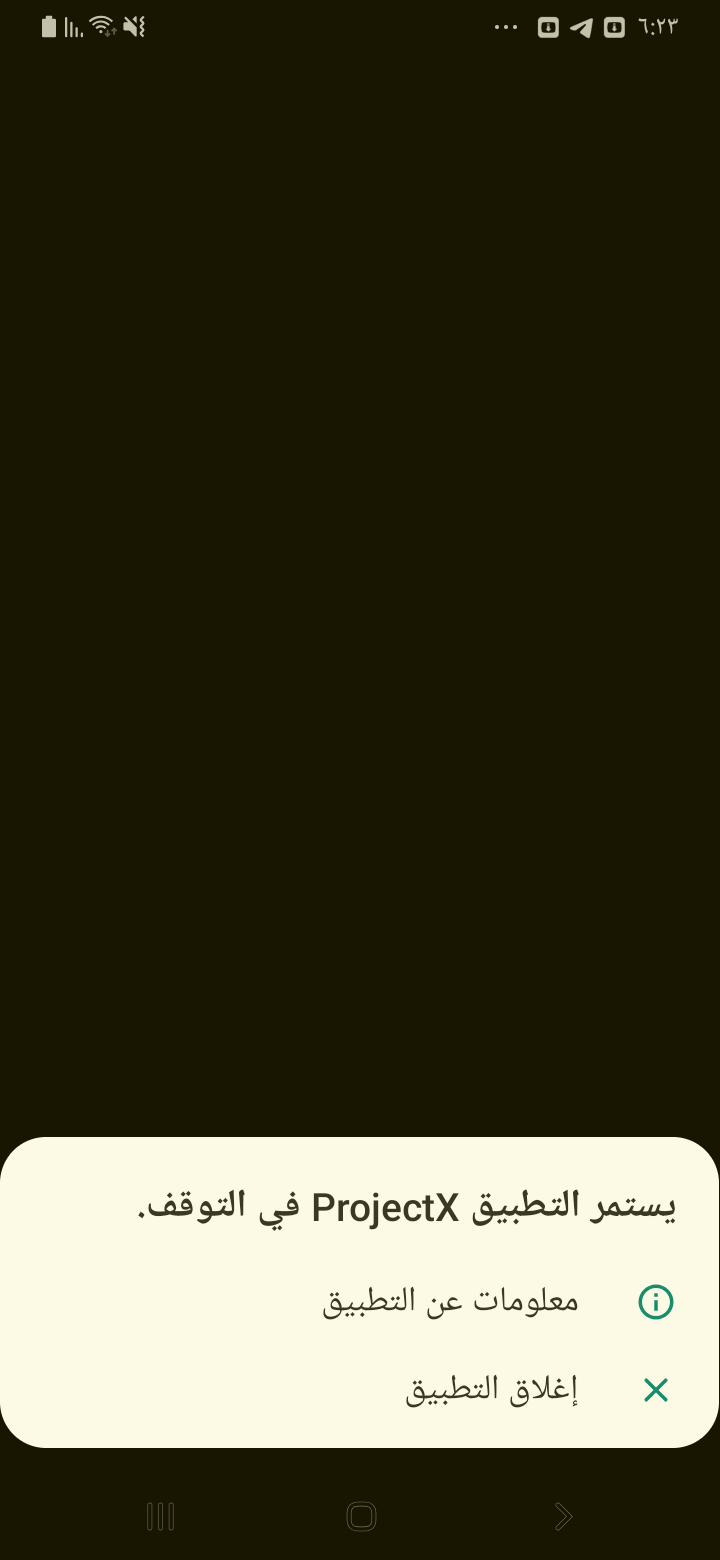
Signs up with no age

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with no email - passed

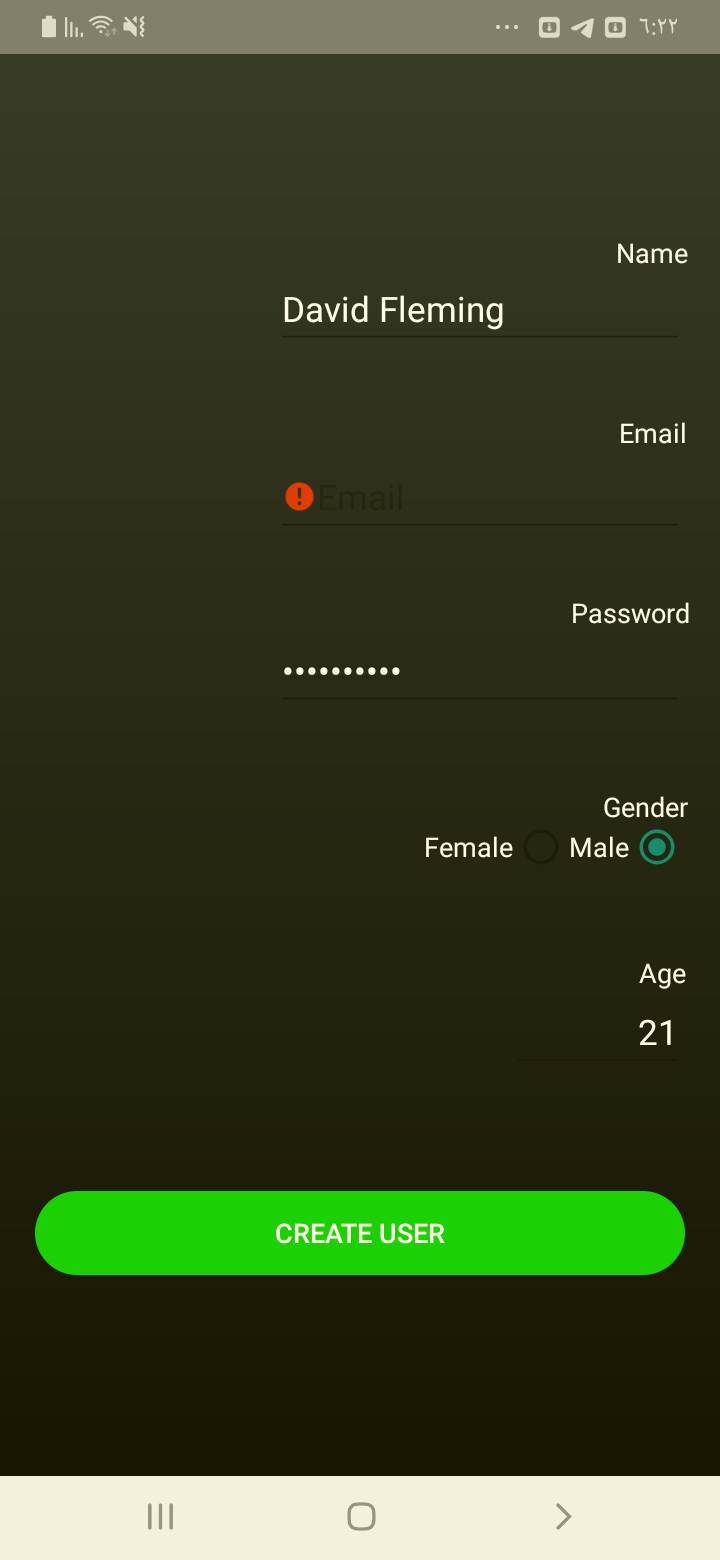
Signs up with no email

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with no name - passed

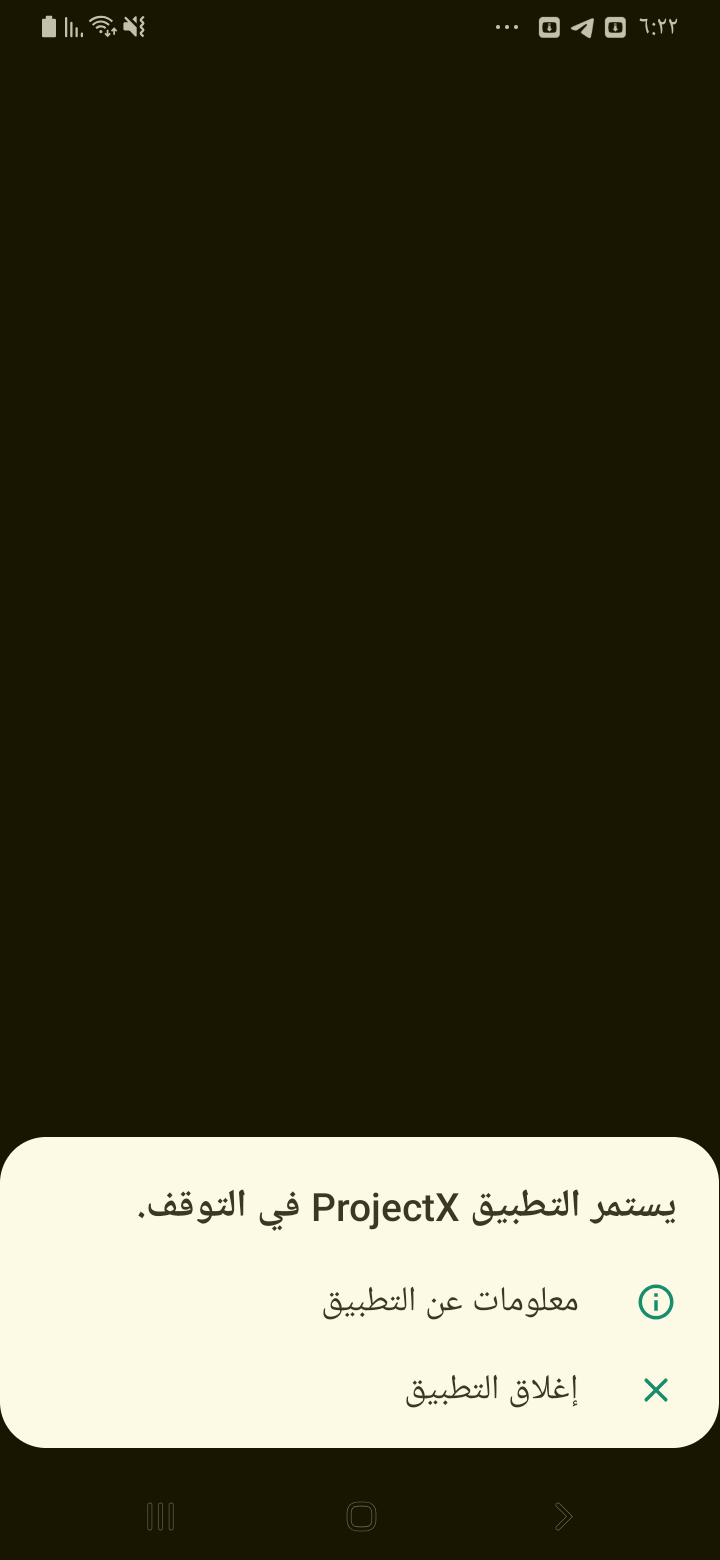
Signs up with no name

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with no password - passed

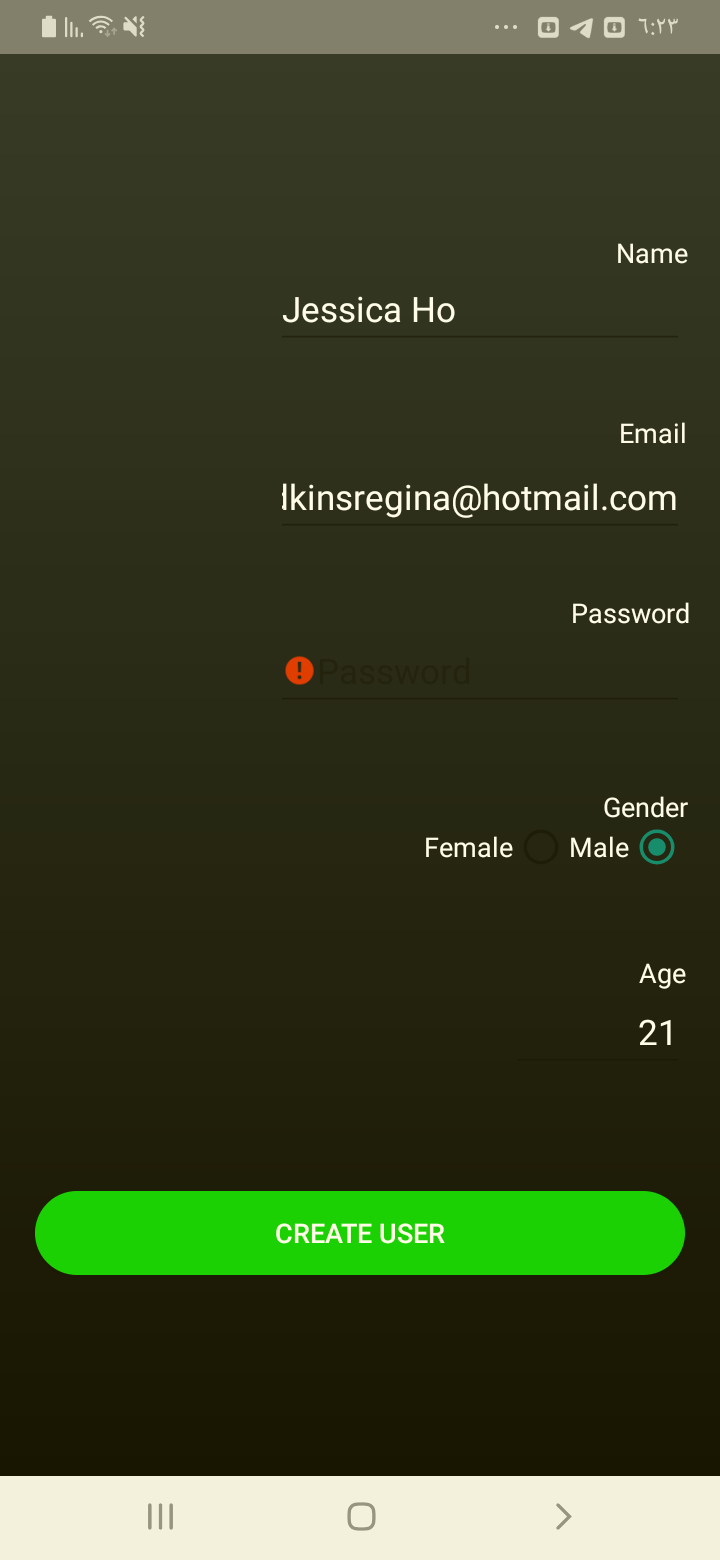
Signs up with no password

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# Signs up with wrong email format - passed

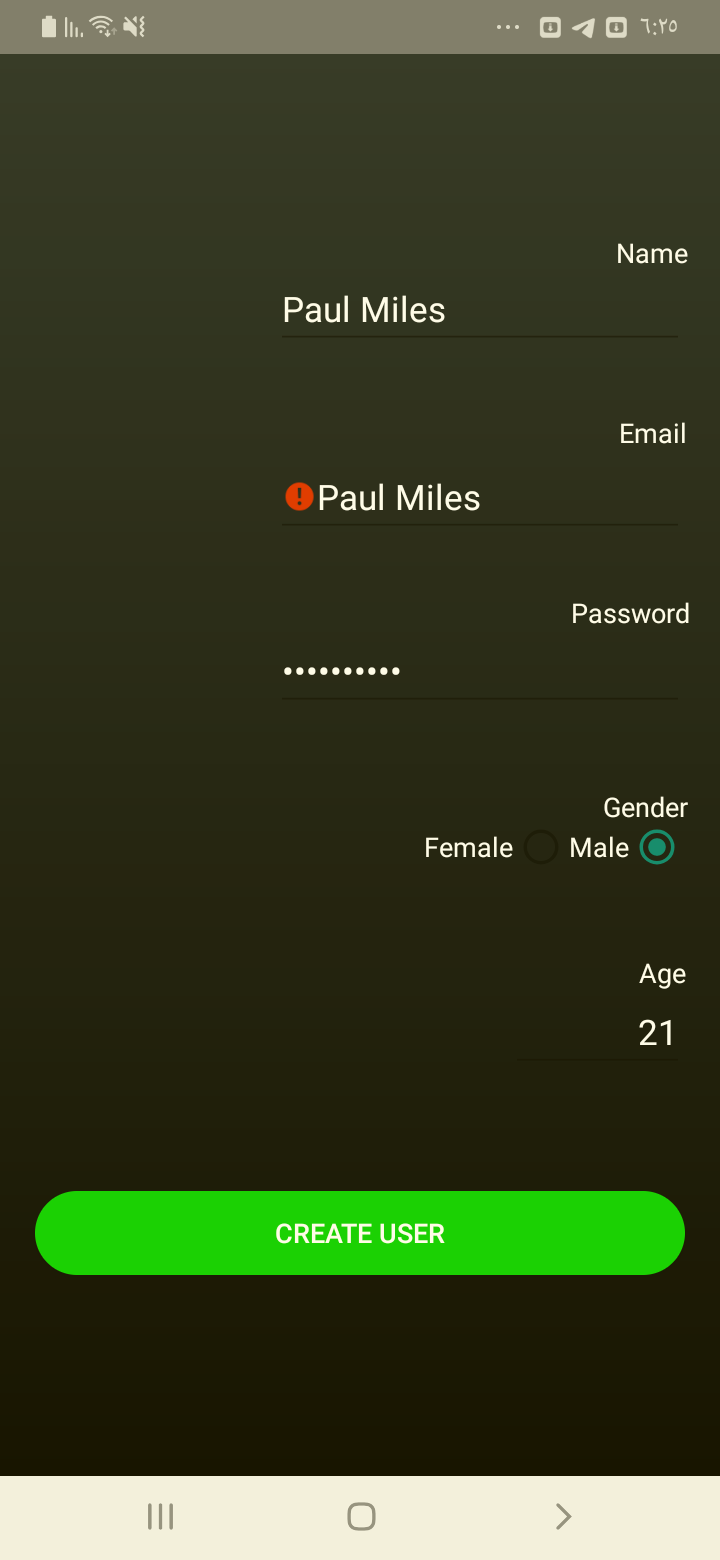
Signs up with wrong email format

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Sign up test Passed



## Test Teardown

[Fixture] setup::0

# checking the artist button - passed

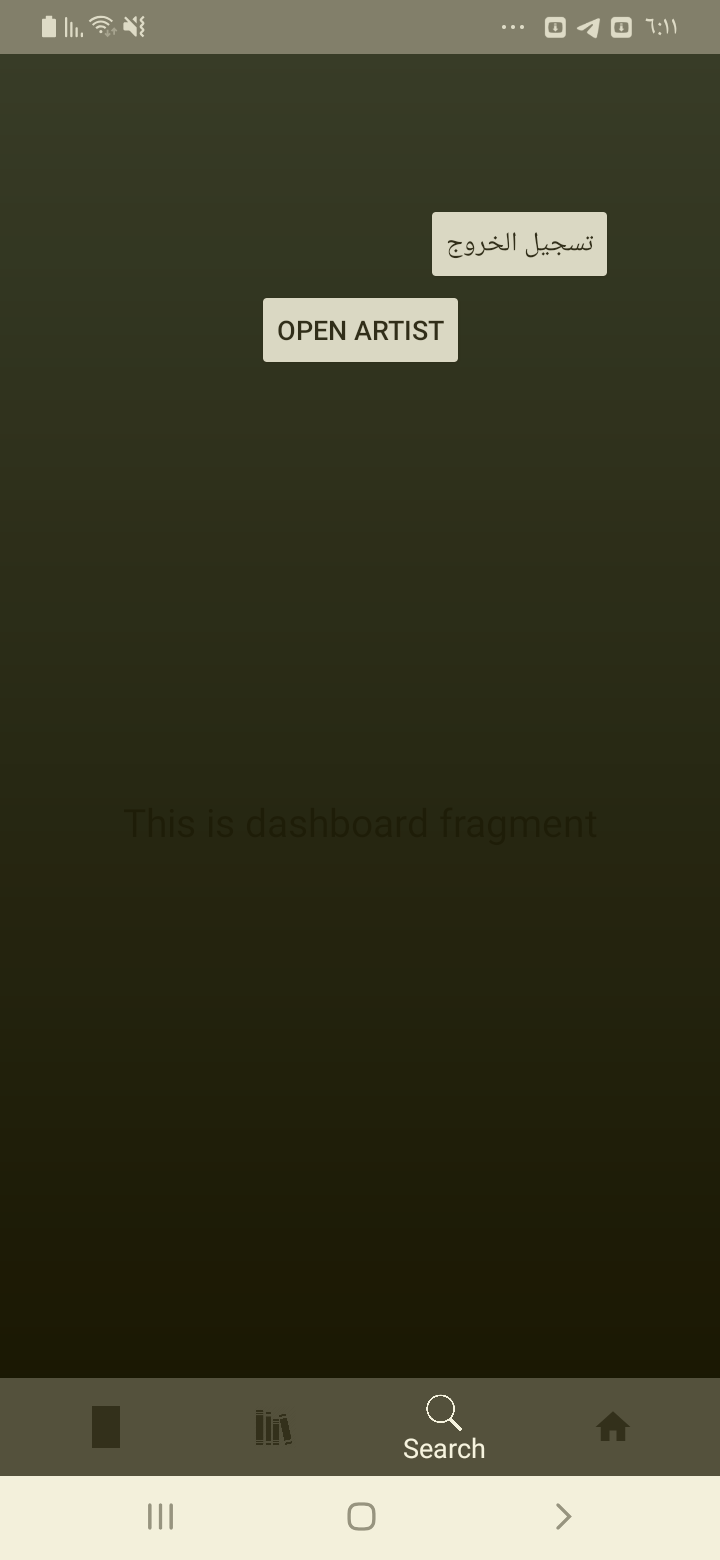
checking the artist button

## Test Setup

[Fixture] setup

## Test Body

[Attachment] Extra test1 passed



[Attachment] Extra test2 passed



[Attachment] Extra test3 passed



## Test Teardown

[Fixture] setup::0