TESTCODERS

Workshop
Robot Framework

Agenda

TESTCODERS

Introduction Tim

- TA Engineer
- Hobbies
- TA Experience
- Companies

Introduction Yuri

- TA Engineer
- Hobbies
- TA Experience
- Companies

Installation

- https://github.com/TestCoders/workshop-robotframework
- NodeJS
- Python
  - Robot Framework
  - Robot Framework Browser Library
- Visual Studio Code
- Extension: Plugin

Installing additional libraries

- Find libraries on github, etc.
- Follow their installation guidelines
- Example:

```
C:\workshop-robotframework> pip install robotframework-requests
```

TESTCODERS

Robot Framework Syntax - Settings

- Importing libraries
- Importing resources
- Setups and Teardowns

```
*** Settings ***
Library     Collections
# Resource     my-keywords.resource

Documentation     Collections Library Documentation:
...               https://robotframework.org/robotframework/latest/libraries/Collections.html
```

TESTCODERS

Robot Framework Syntax – Variables

- Predefined values
- Global scope
- $, @, & combined with {}

```
*** Variables ***
${SCALAR}     Martijn
@{LIST}     Tim     Mark     Yuri
&{DICTIONARY}     name=Tim     email=tim@testcoders.nl     age=512
```

TESTCODERS

Robot Framework Syntax – Test Cases

- Test name
- Indentation and whitespaces
- Keyword Arguments

```
*** Test Cases ***
Log Some Data
    [Documentation]    This is test documentation
    # This is a comment
    Log    message=Data

Log With A Keyword
    Log Data With A Keyword
```

TESTCODERS

Robot Framework Syntax – Keywords

- Same structure as tests
- Same keyword usage
- Keywords contain other keywords

```
*** Test Cases ***
Log With A Keyword
    Log Data With A Keyword

*** Keywords ***
Log Data With A Keyword
    [Documentation]    This is keyword documentation
    Log    Some Data
    Log    message=${SCALAR}
    Log List    list_=${LIST}
    Log Dictionary    dictionary=${DICTIONARY}    level=WARN
```

TESTCODERS

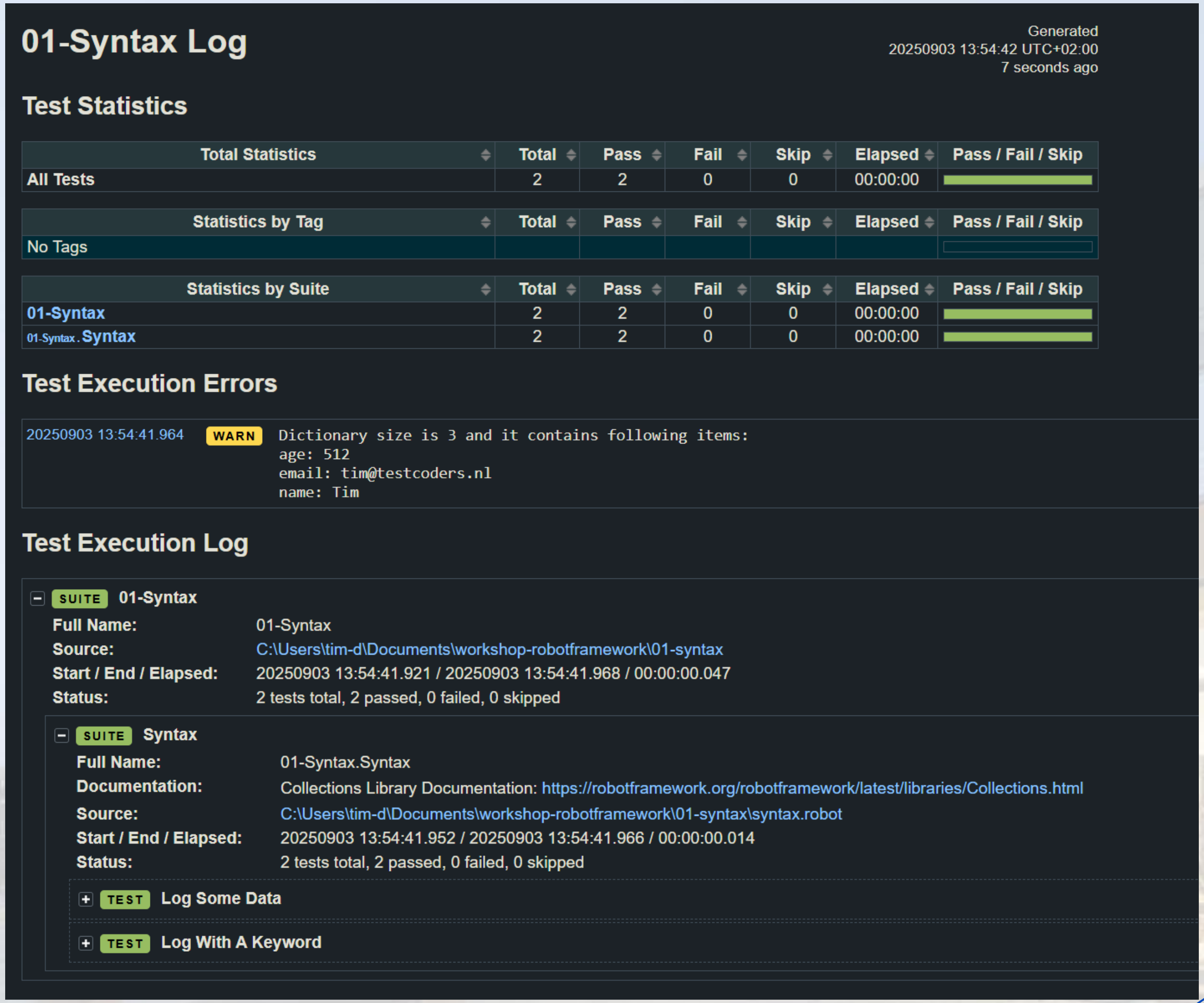
Robot Framework Syntax – Keywords

- Same structure as tests
- Same keyword usage
- Keywords contain other keywords

```
*** Test Cases ***
Log With A Keyword
    Log Data With A Keyword

*** Keywords ***
Log Data With A Keyword
    [Documentation]    This is keyword documentation
    Log    Some Data
    Log    message=${SCALAR}
    Log List    list_=${LIST}
    Log Dictionary    dictionary=${DICTIONARY}    level=WARN
```

TESTCODERS

Robot Framework Syntax – Log



01-Syntax Log

Generated
20250903 13:54:42 UTC+02:00
7 seconds ago

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	2	2	0	0	00:00:00	🟩

Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
No Tags						

Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
01-Syntax	2	2	0	0	00:00:00	🟩
01-Syntax . Syntax	2	2	0	0	00:00:00	🟩

Test Execution Errors

```
20250903 13:54:41.964  WARN  Dictionary size is 3 and it contains following items:
                             age: 512
                             email: tim@testcoders.nl
                             name: Tim
```

Test Execution Log

SUITE 01-Syntax

Full Name:	01-Syntax
Source:	C:\Users\tim-d\Documents\workshop-robotframework\01-syntax
Start / End / Elapsed:	20250903 13:54:41.921 / 20250903 13:54:41.968 / 00:00:00.047
Status:	2 tests total, 2 passed, 0 failed, 0 skipped

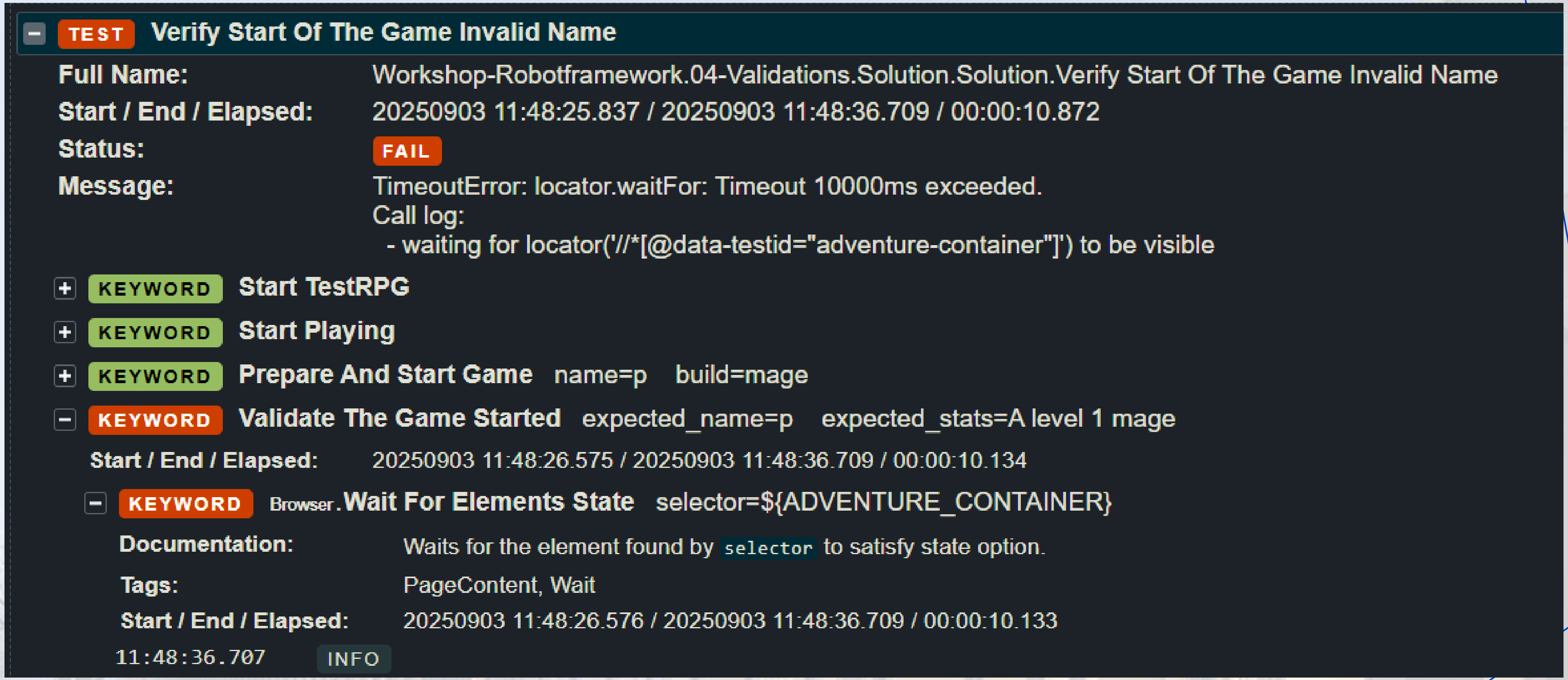
SUITE Syntax

Full Name:	01-Syntax.Syntax
Documentation:	Collections Library Documentation: https://robotframework.org/robotframework/latest/libraries/Collections.html
Source:	C:\Users\tim-d\Documents\workshop-robotframework\01-syntax\syntax.robot
Start / End / Elapsed:	20250903 13:54:41.952 / 20250903 13:54:41.966 / 00:00:00.014
Status:	2 tests total, 2 passed, 0 failed, 0 skipped

TEST Log Some Data

TEST Log With A Keyword

Robot Framework Syntax – Log



```
TEST  Verify Start Of The Game Invalid Name
  Full Name:               Workshop-Robotframework.04-Validations.Solution.Solution.Verify Start Of The Game Invalid Name
  Start / End / Elapsed:   20250903 11:48:25.837 / 20250903 11:48:36.709 / 00:00:10.872
  Status:                  FAIL
  Message:                 TimeoutError: locator.waitFor: Timeout 10000ms exceeded.
                           Call log:
                             - waiting for locator('//*[@data-testid="adventure-container"]') to be visible
  + KEYWORD  Start TestRPG
  + KEYWORD  Start Playing
  + KEYWORD  Prepare And Start Game   name=p   build=mage
  - KEYWORD  Validate The Game Started   expected_name=p   expected_stats=A level 1 mage
      Start / End / Elapsed:   20250903 11:48:26.575 / 20250903 11:48:36.709 / 00:00:10.134
    - KEYWORD  Browser.Wait For Elements State   selector=${ADVENTURE_CONTAINER}
        Documentation:           Waits for the element found by selector to satisfy state option.
        Tags:                    PageContent, Wait
        Start / End / Elapsed:   20250903 11:48:26.576 / 20250903 11:48:36.709 / 00:00:10.133
      11:48:36.707   INFO
```

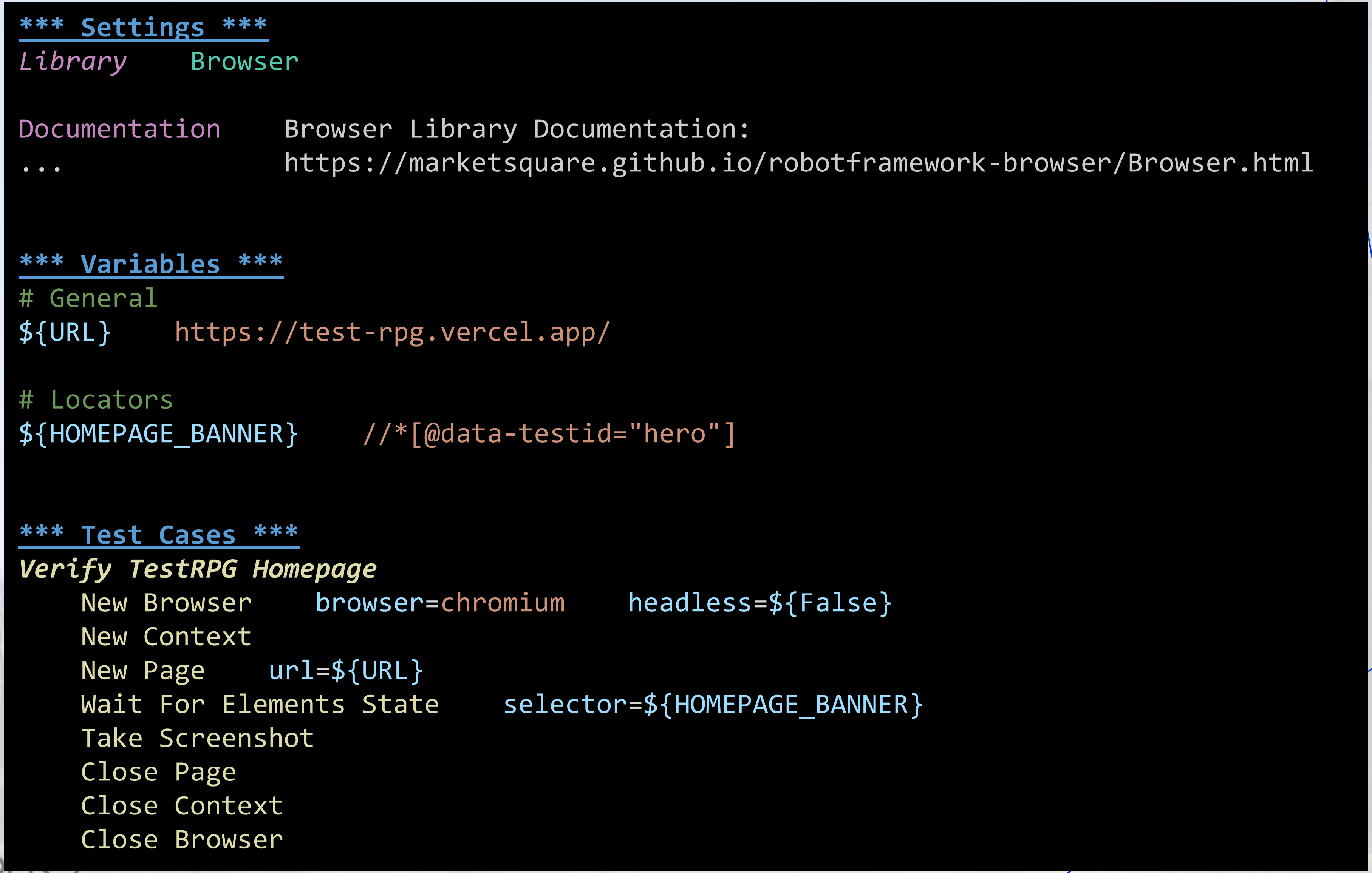
TESTCODERS

Robot Framework Syntax – Documentation

- Builtin Library
  https://robotframework.org/robotframework/latest/libraries/BuiltIn.html
- General library information and usage
- Keywords
- Mandatory and optional arguments
- Examples

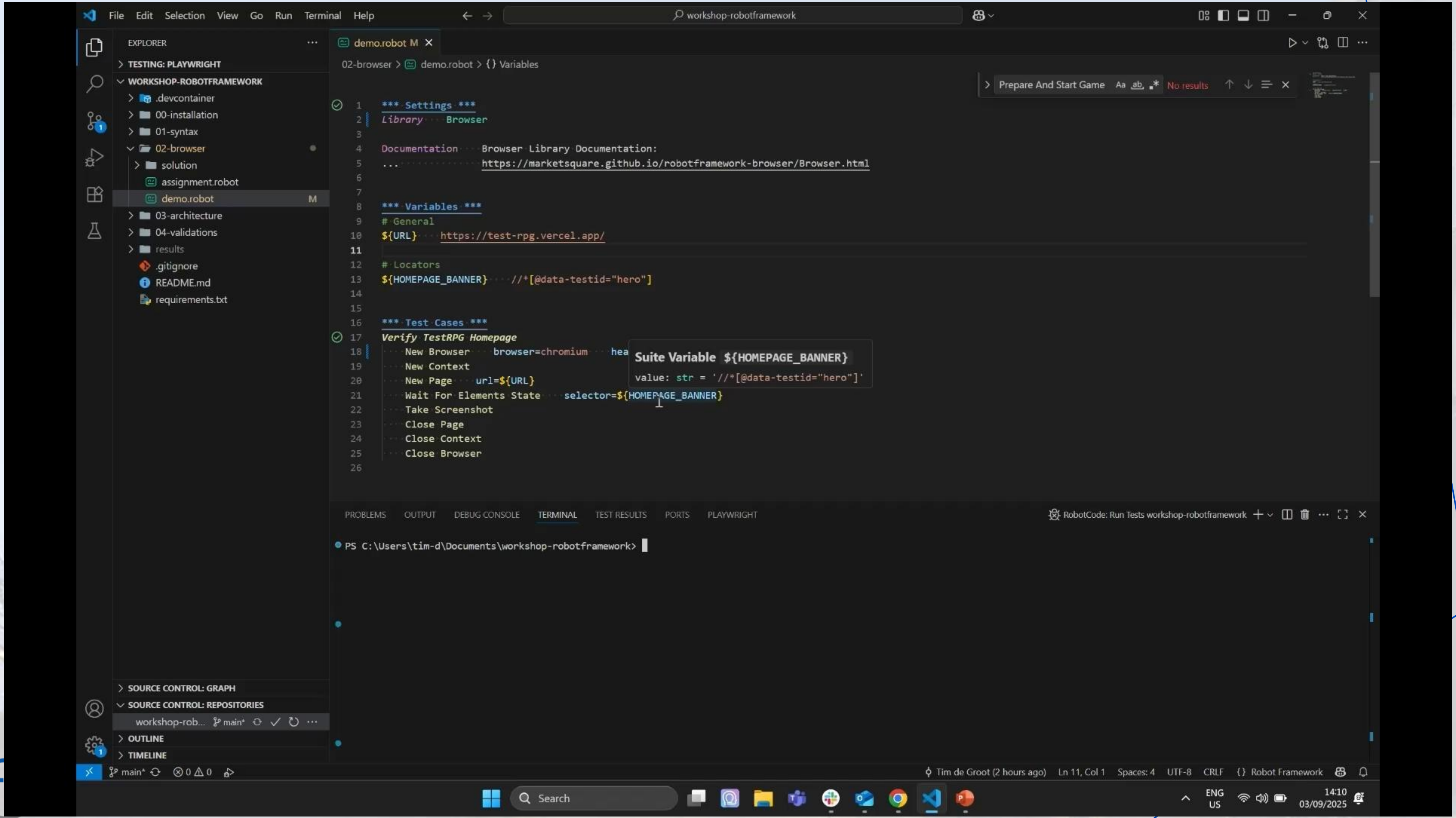
TESTCODERS

Browser Testing



```
*** Settings ***
Library     Browser

Documentation     Browser Library Documentation:
...               https://marketsquare.github.io/robotframework-browser/Browser.html


*** Variables ***
# General
${URL}     https://test-rpg.vercel.app/

# Locators
${HOMEPAGE_BANNER}     //*[@data-testid="hero"]


*** Test Cases ***
Verify TestRPG Homepage
    New Browser     browser=chromium     headless=${False}
    New Context
    New Page     url=${URL}
    Wait For Elements State     selector=${HOMEPAGE_BANNER}
    Take Screenshot
    Close Page
    Close Context
    Close Browser
```

Browser Testing



```
*** Settings ***
Library    Browser

Documentation    Browser Library Documentation:
...              https://marketsquare.github.io/robotframework-browser/Browser.html


*** Variables ***
# General
${URL}    https://test-rpg.vercel.app/

# Locators
${HOMEPAGE_BANNER}    //*[@data-testid="hero"]


*** Test Cases ***
Verify TestRPG Homepage
    New Browser    browser=chromium    hea
    New Context
    New Page    url=${URL}
    Wait For Elements State    selector=${HOMEPAGE_BANNER}
    Take Screenshot
    Close Page
    Close Context
    Close Browser
```

```
Suite Variable ${HOMEPAGE_BANNER}
value: str = '//*[@data-testid="hero"]'
```

```
PS C:\Users\tim-d\Documents\workshop-robotframework>
```

Browser Testing – Assignment

- See 02-browser/assignment.robot
- Add the required keywords to complete the testcase
  - 1. Start playing
  - 2. Enter a username
  - 3. (Optionally) Select a different build
  - 4. Start the game

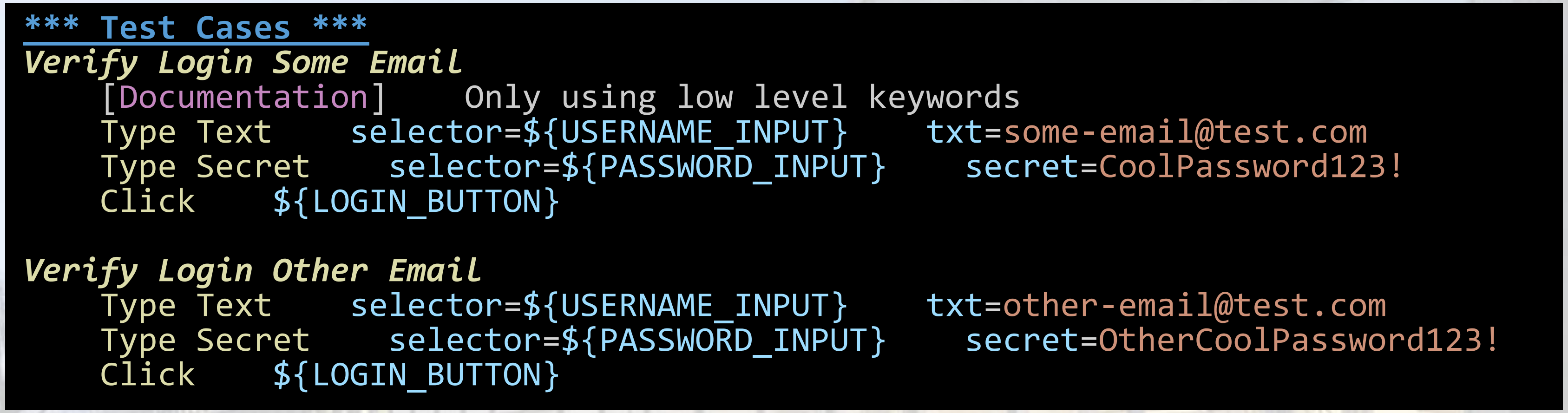
TESTCODERS

Browser Testing – Assignment Solution

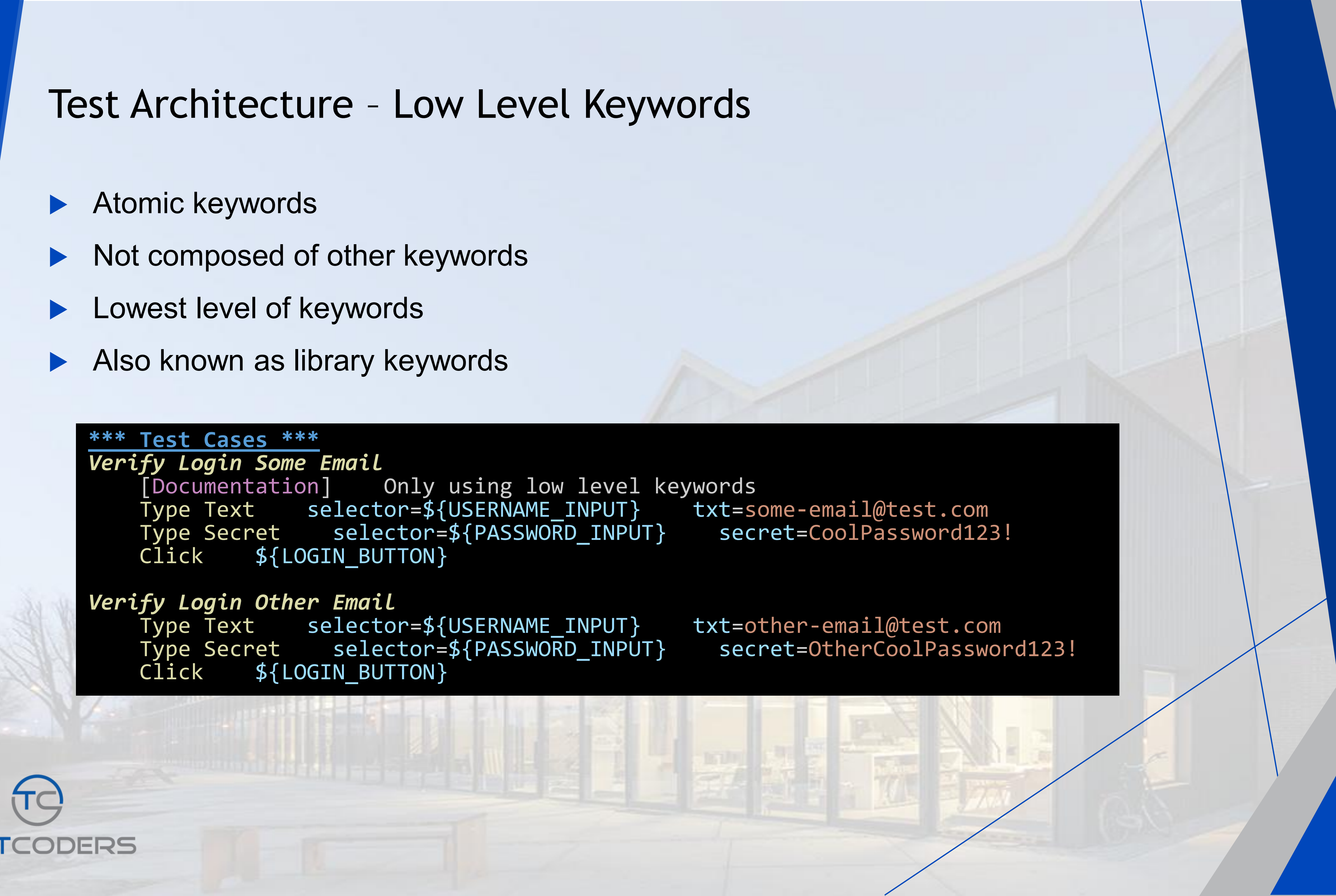
```
*** Test Cases ***
Verify Start Of The Game
    New Browser    browser=chromium    headless=${False}
    New Context
    New Page    url=${URL}
    # Click "Click here to play"
    Click    selector=${CLICK_HERE_TO_PLAY_BUTTON}
    # Enter your character name
    Type Text    selector=${CHARACTER_NAME_INPUT}    txt=Tim
    # (Optional) Select your build
    Select Options By    ${BUILD_SELECT}    value    mage
    # Click "Start!"
    Click    selector=${START_GAME_BUTTON}
    Close Page
    Close Context
    Close Browser
```

Test Architecture – Low Level Keywords

- Atomic keywords
- Not composed of other keywords
- Lowest level of keywords
- Also known as library keywords



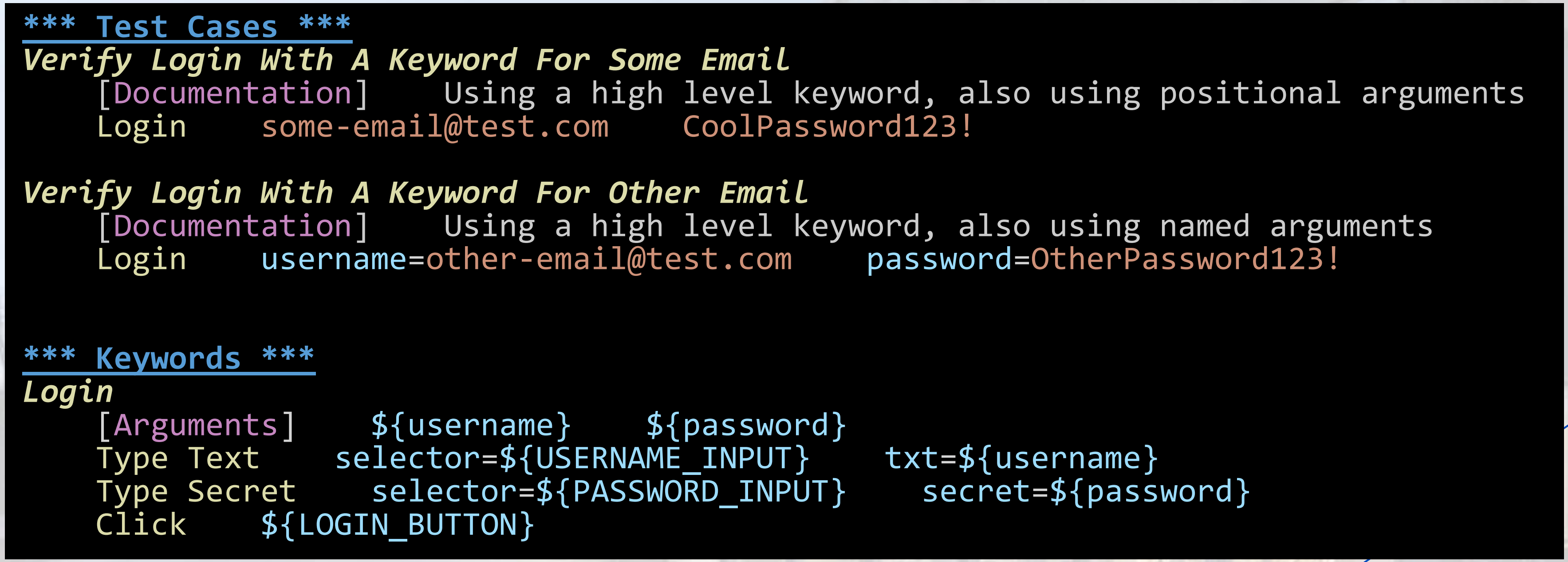
```
*** Test Cases ***
Verify Login Some Email
    [Documentation]    Only using low level keywords
    Type Text    selector=${USERNAME_INPUT}    txt=some-email@test.com
    Type Secret    selector=${PASSWORD_INPUT}    secret=CoolPassword123!
    Click    ${LOGIN_BUTTON}

Verify Login Other Email
    Type Text    selector=${USERNAME_INPUT}    txt=other-email@test.com
    Type Secret    selector=${PASSWORD_INPUT}    secret=OtherCoolPassword123!
    Click    ${LOGIN_BUTTON}
```



TESTCODERS

Test Architecture – High Level Keywords

- Composite keywords
- Composed of other keywords
- Used for structuring tests
- Also known as user keywords



```
*** Test Cases ***
Verify Login With A Keyword For Some Email
    [Documentation]    Using a high level keyword, also using positional arguments
    Login    some-email@test.com    CoolPassword123!

Verify Login With A Keyword For Other Email
    [Documentation]    Using a high level keyword, also using named arguments
    Login    username=other-email@test.com    password=OtherPassword123!


*** Keywords ***
Login
    [Arguments]    ${username}    ${password}
    Type Text    selector=${USERNAME_INPUT}    txt=${username}
    Type Secret    selector=${PASSWORD_INPUT}    secret=${password}
    Click    ${LOGIN_BUTTON}
```

Test Architecture – Assignment

- See 03-architecture/assignment.robot
- Create the required high level keywords to structure the testcase

TESTCODERS

Test Architecture – Assignment Solution

```
*** Test Cases ***
Verify Start Of The Game
    Start TestRPG
    Start Playing
    Prepare And Start Game    name=Tim    build=mage
    Close TestRPG

*** Keywords ***
Start TestRPG
    New Browser    browser=chromium    headless=${False}
    New Context
    New Page    url=${URL}

Start Playing
    Click    selector=${CLICK_HERE_TO_PLAY_BUTTON}

Prepare And Start Game
    [Arguments]    ${name}    ${build}
    Type Text    selector=${CHARACTER_NAME_INPUT}    txt=${name}
    Select Options By    ${BUILD_SELECT}    value    ${build}
    Click    selector=${START_GAME_BUTTON}

Close TestRPG
    Close Page
    Close Context
    Close Browser
```

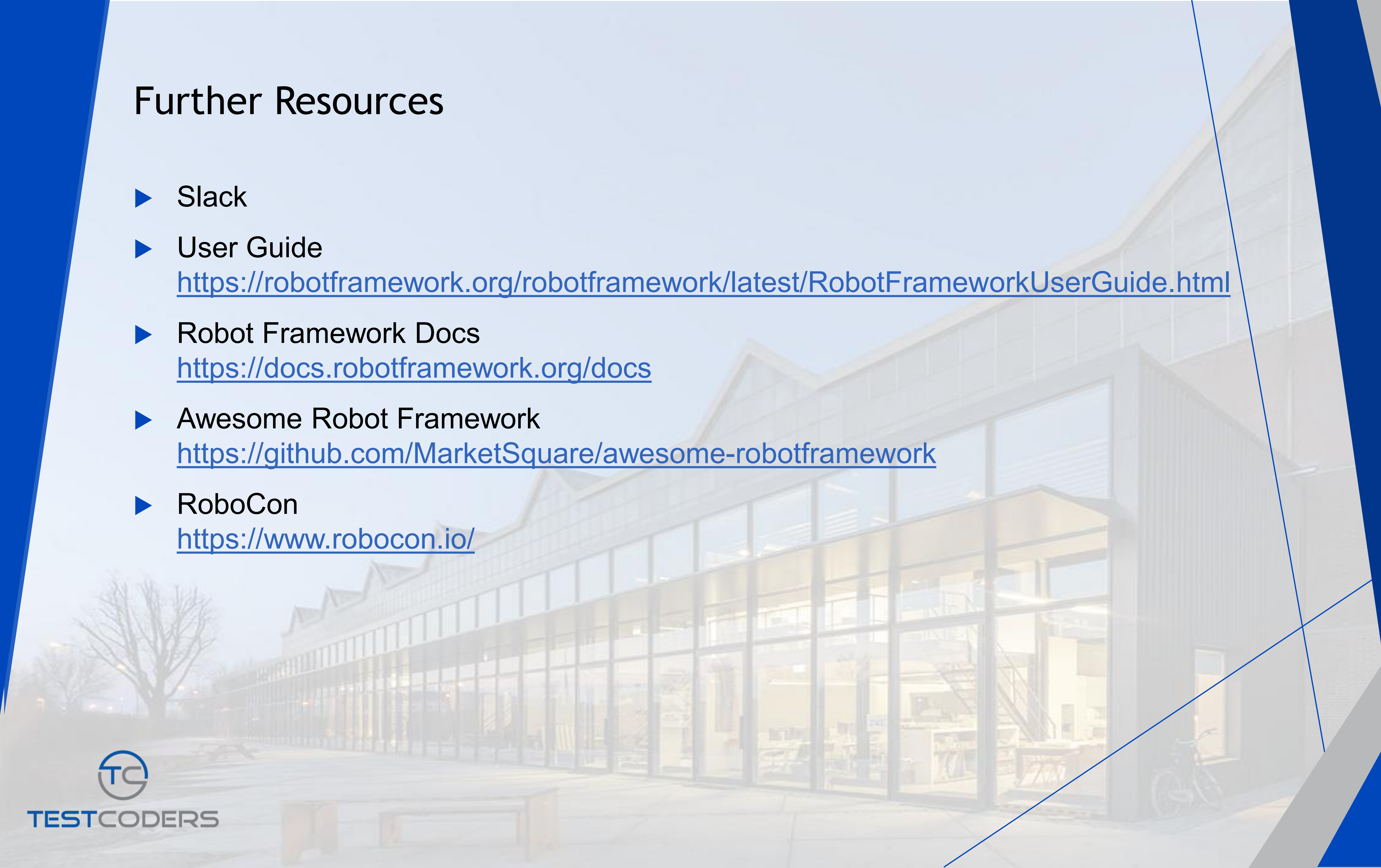
Validations

```
*** Test Cases ***
Verify Start Of The Game
    Start TestRPG
    Start Playing
    Prepare And Start Game    name=z    build=mage
    Close TestRPG
```

Validations

- Validate positives!

```
*** Test Cases ***
Verify Start Of The Game Valid Name
    Start TestRPG
    Start Playing
    Prepare And Start Game    name=Pietje    build=mage
    Validate The Game Started    expected_name=Pietje    expected_stats=A level 1 mage
    Close TestRPG

Verify Start Of The Game Invalid Name
    Start TestRPG
    Start Playing
    Prepare And Start Game    name=p    build=mage
    Validate The Game Started    expected_name=p    expected_stats=A level 1 mage
    Close TestRPG


*** Keywords ***
Validate The Game Started
    [Arguments]    ${expected_name}    ${expected_stats}
    Wait For Elements State    selector=${ADVENTURE_CONTAINER}
    ${actual_name}    Get Text    selector=${CHARACTER_NAME_LABEL}
    Should Be Equal As Strings    first=${expected_name}    second=${actual_name}
    ${actual_stats}    Get Text    selector=${CHARACTER_STATS_LABEL}
    Should Be Equal As Strings    first=${expected_stats}    second=${actual_stats}
```



Further Resources

- Slack
- User Guide
  https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html
- Robot Framework Docs
  https://docs.robotframework.org/docs
- Awesome Robot Framework
  https://github.com/MarketSquare/awesome-robotframework
- RoboCon
  https://www.robocon.io/

Recap

- Installation
- Syntax
- Browser testing
- Architecture
- Validations

- Questions?

TESTCODERS

API Testing

```
*** Settings ***
Library     Collections
Library     RequestsLibrary

Documentation    Documentation: https://github.com/MarketSquare/robotframework-requests


*** Variables ***
${URL}    https://test-rpg.vercel.app/api


*** Test Cases ***
Validate Builds
    ${response}    GET    url=${URL}/builds
    Dictionary Should Contain Key    dictionary=${response.json()}    key=thief
    Dictionary Should Contain Key    dictionary=${response.json()}    key=mage
    Dictionary Should Contain Key    dictionary=${response.json()}    key=knight
    Dictionary Should Contain Key    dictionary=${response.json()}    key=brigadier
```

TESTCODERS

Python Keywords

keywords.py

```
import random

def get_random_build_from_list(list):
    result = random.choice(list)
    print(f'Input: {list}, result: {result}')
    return result
```

python.robot

```
*** Settings ***
Library     keywords.py

*** Variables ***
@{BUILDS}     Thief     Mage     Knight     Brigadier

*** Test Cases ***
Get Random Build
    ${build}     Get Random Build From List     list=${BUILDS}
    Should Contain     container=${BUILDS}     item=${build}
```

TESTCODERS