FQA Challenge



## Description

The QA onboarding challenge for a FQA Engineer is divided in three main parts:

Create a test plan document simulating a real project to expose the new hire to a real project’s experience planning and defining the testing steps and deliverables required.

The presentation is aimed to make the mentee experiment a role play making a proposal with the client on a testing architecture, explaining What and Why was it designed that way.

Last but not least, a technical challenge for frontend and backend creating and delivering test cases.

## Objectives

* Successfully create a test plan document for a project defining the following details:
  + Defining which are the test items.
  + Testing Scope:
    - In Scope
    - Out of Scope
    - Browsers
    - Operating Systems
    - Testing Types
    - Testing tools
  + Testing flow.
  + Defect management.
  + Testing/Development process overview.
  + Definition of done
  + Definition of Ready
  + Automated validation process flow (Optional)
  + Metrics
* Create a presentation for the business audience to explain the QA process defined on the test plan. This should be an executive presentation in English that summarizes all the elements in the test strategy and you should be able to present it in very high level language so people with no technical background can understand the importance of the strategy and its elements. We need to use one of Wizeline’s templates for presentations.

**Duration**: 45 min (Introduction, presentation, Q&A and, feedback)

* Deliver the technical challenge based on the simulated project.

## Project details

**Todoist** is the productivity tool you need to get work and life organized. Collect tasks, organize projects, and plan your day.

**Site**: [https://todoist.com/](https://todoist.com/home)

**Environments**: Dev and Prod

**Frontend:** Angular

**Backend:** REST

#### **Frontend challenge**

Navigate to [Todoist](https://todoist.com/) and create an account, get familiar with the site since we are using it for the whole challenge.

Create test cases into a google sheet that must be properly detailed and documented.

* Successful login - Define a test case that performs a successful login
* Unsuccessful login - Define multiple negative scenarios for login.
* Create a new task - Create a new task and validate it was created correctly
* Create a new project - Create a new project and validate it was created correctly

### 

#### **Backend challenge**

Download [Postman](https://www.getpostman.com/) to create a new collection and a new environment for your variables and navigate to [Todoist API](https://developer.todoist.com/rest/v1/). ​​You can find your personal token in the integrations settings view of the Todoist web app and replace the token value in your requests, once you have installed and configured your collection please follow the following steps

1. Get your authorization token and save it as an environment variable
2. Create a new folder inside your collection for projects and another one for tasks.
3. Create the following endpoints and its corresponding Tests: status codes, content, response time, etc.
   * Projects
     1. Get all projects
     2. Create a new project
     3. Update a project
     4. Delete a project
   * Tasks:
     1. Get active tasks
     2. Create a new task
     3. Get an active task
     4. Update a task
     5. Delete a task

Optional:

1. Integrate one framework for API Test Automation i.e. \*[Newman](https://www.npmjs.com/package/newman)

## References

Backend:

* [Postman](https://www.getpostman.com/)
* [Newman](https://www.npmjs.com/package/newman)

Test Strategies:

* [Test Plan](https://docs.google.com/document/d/1UESEDf3h1_pW6ZSWkynOk1yS_muBcyCOnohPpO1chKo/edit?usp=sharing)