Test Plan Document

IDENTIFICATION INFORMATION SECTION

PRODUCT

- **Product Name:** Todoist

FEATURE DESCRIPTION

- Item Management Feature: adding, updating, and completing a todo item

Test engineers

- Nathan Nguyen

Backend engineers:

- N/A

Frontend engineers:

- N/A

UNIT TEST SECTION

UNIT TEST STRATEGY / EXTENT OF UNIT TESTING:

Evaluate new features and bug fixes introduced in this release

UNIT TEST CASES

N/A

REGRESSION TEST SECTION

- Execute all current tests (automation/manual) which affected by this new feature
- Smoke test on prod after prd deploy

API TEST STRATEGY

- ### Approach
  - API document should send to SDET
  - SDETs develop test script
  - SDETs send test script to Backend engineers and push to test repo for CI
  - Backend engineer run scripts on local machine after finish developing
  - Backend engineer request a PR to CI to trigger test on uat/staging environment
- ### Testing tool:
  - Postman :
    - Common tool shared by team
    - Able to share between SDET <> BE <> FE
  - Docker :
    - No install required to execute test
    - CI friendly

API TEST CASES

#	OBJECTIVE	INPUT	EXPECTED RESULTS
1	User is able to create a new todo item	todoItemContent, todoItemDueDate	SttCode: 200 / item return with input data
2	User is not able to create a new todo item with invalid dueDate	todoItemContent, todoItemDueDate(incorrect format (`hgiyfif`)	SttCode: 400 / Res body: Date is invalid
3	User is not able to create a new todo item with empty itemContent	todoItemContent, todoItemContent = `empty`	SttCode: 400 / Res body: Empty content

#	OBJECTIVE	INPUT	EXPECTED RESULTS
4	User is able to edit an existed todo item	todoItemContent, itemID which exist	SttCode: 204 / item return with input data through get todoItem
5	User is not able to edit an non-existed todo item	todoItemContent, itemID which not exist	SttCode: 400 / Res body: Item not found
6	User is not able to edit a todo item with empty content	todoItemContent=`empty`, itemID which exist	SttCode: 400 / Res body: Item not found
7	User is able to close an existed todo item	itemID which exist	SttCode: 204 / item will be not found with input itemID through get todoItem
8	`User is able to close an non-existed todo item?`	`itemID`	`? Prod behavior is different than my expect`

End to End TEST SECTION

End to end test through browser to verify user behavior

End to End TEST STRATEGY AND EXTENT OF End to End TESTING

- Approach
  - Start develop right after developing api test finished
  - Validate user's interactions from browser
  - Make sure each action (CRUD) will send `sync` request to backend to save data to system
  - Make sure data display as expected

- Testing tool:
  - A Custom NodeJS test framework
  - Integrated with Protract `https://www.protractortest.org/#/` (Jest / Wdio are ok too)
  - CucumberJS for BDD test case
  - Report Portal for reporting
  - Slack notifier for reporting on slack
  - Docker: CI Friendly, stable browser environment

End to End TEST CASES

`ALl of those test cases requires to check sync request to backend as expected`

#	OBJECTIVE	INPUT	EXPECTED RESULTS
1	User is able to create a new todo item	todoItemContent, todoItemDueDate	Item display on item list with correct input data
2	User is able to edit an existed todo item	todoItemContent	Item display on item list with correct input data
3	User is able to close an existed todo item	todoItemContent	Item should be disappeared on the list, Task done notifier should show