

# MSD RQ - Assignment 5

## MSD Reading Questions self-assessment bot

### Team members

Kristina Alyabeva - Project manager

Saveliy Chertkov - Developer

Kirill Korikov - Scrum master

[Repository on GitLab](#)

### Review with the customer

☒ ~~Show you v1 version to the customer~~

Unfortunately, this week we rescheduled a meeting with the customer and it will take place only on 11.14.2024

We sent him the changes containing:

- Checking for examples
- Changing the response structure

☒ ~~Show this version of the product to your customer. Point out any new features that you implemented. Confirm that you understood everything correctly.~~

☒ ~~Document all feedback.~~

☒ ~~Describe what features you are planning to implement next week.~~

~~This week we plan to work as agreed with the customer:~~

- Implement a check for all issues
- Connect keys to gemini to implement other functions
- Get feedback from students on UX

### Add unit tests

☒ ~~Add appropriate unit test framework for your project. For example, if you are using Python, I recommend pytest. If you are using Go, pick an appropriate test framework for Go.~~

We use pytest module for the application testing.

☒ ~~Add test coverage.~~

- ☒ Add unit tests. You should aim for at least 50% coverage. The test coverage should not be less than 30% for this assignment.

Our Unit-tests cover 54% of the code.

```
----- coverage: platform win32, python 3.12.0-final-0 -----
```

Name	Stmts	Miss	Cover
-----			
MainTask\Bot\__init__.py	0	0	100%
MainTask\Bot\botDef.py	40	40	0%
MainTask\PDFModule\PDFChecker.py	75	41	45%
MainTask\PDFModule\PDFFile.py	51	0	100%
MainTask\PDFModule\PDFTest\PDFChecker_test.py	18	0	100%
MainTask\PDFModule\PDFTest\PDF_test.py	13	0	100%
MainTask\PDFModule\PDFTest\__init__.py	0	0	100%
MainTask\PDFModule\__init__.py	0	0	100%
MainTask\__init__.py	0	0	100%
MainTask\main.py	16	16	0%
-----			
TOTAL	213	97	54%

- ☒ Add documentation on how to run your tests and check the test coverage.

Test documentation you can find in [repository](#).

## Extra credit opportunity

- ☐ Set up production and staging application instances.
- ☒ Set up continuous integration and continuous deployment pipelines.

We have only done the CI part.

There is no CD part in our project, because our project is not deployed on the server.

The CI part contains the build and test stages.

You can check it out in the [repository](#).

## Sprint cycle

- ☒ Review changes that you were planning to add to this sprint. Make sure you account for

~~adding unit tests this sprint.~~

[Initial plan](#) had a mistake: requirement specified making 5 personas, not 10.

Our team couldn't contact customer for a review this week, so instead we sent them screenshots of the current project state.

- ✓ ~~Review your strategic plan. Are you still on track? Does it need to be modified? Update your plan if needed.~~

[Iteration 2](#) is almost fully done. Only thing left is to check the relevance of an answer which is harder than anticipated and will be moved to the next iteration.

- ✓ ~~Review your sprint process and determine what can be improved. Look at the results of your previous sprint retrospective. Decide how you are going to improve this sprint based on that retrospective.~~

Current sprint processes fully satisfy the team, so we don't see any need in changing anything.

- ✓ ~~Conduct sprint planning.~~
  - Implement feature of checking presence of all answers, researching methods to use AI to inspect answer's relevance.
  - Conduct 2 demonstrations of the bot to end-users to gather their feedback
- ✓ ~~Describe the sprint goal.~~

Completion of [Iteration 3](#) features.

- ✓ ~~Determine what tasks need to be done and who is going to do those tasks.~~

Apart from this assignment, there were no management tasks needed during this sprint, next week is uncertain until the next assignment appears.

For now, **Saveliy** implements new features in the bot.

**Kristina** manages overall work, curating tasks priority.

**Kirill** conducts interviews with the end-users

- ✓ ~~Execute the sprint. There should be completed sprint artifacts like tasks, code, merge requests, etc.~~
- [Link to a Gitlab](#)
- ✓ ~~Conduct a sprint retrospective. Describe the results of the retrospective.~~

**What went well:** During that sprint we've learned that feedback from the end-users is highly valuable and had to be collected at the very start.

**What went wrong:** this week's workload didn't allow for an effective communication, so sprint

was done mostly blindly.

**Possible improvements:** we still lack communication