
Software Requirements Specification

AYU plug-in

Version 1.0

Authors - Avi Boter , Yair Anisfeld, Uriel Sachs

10/12/2020

Table of Contents

Table of Contents	ii
Revision History	שגיאה! הסימניה אינה מוגדרת.
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Project Scope	1
1.3 References.....	1
2. Overall Description	2
2.1 Product Perspective.....	2
2.2 Product Features	2
2.3 Operating Environment.....	2
2.4 Design and Implementation Constraints	3
3. System Features	4
3.1 Subbmision Box.....	שגיאה! הסימניה אינה מוגדרת.
3.2 FeedBack Window.....	שגיאה! הסימניה אינה מוגדרת.
4. Other Nonfunctional Requirements.....	5
4.1 Performance Requirements	5
4.2 Security Requirements	5
4.3 Software Quality Attributes	5

1. Introduction

1.1 Purpose

Moodle system is an online academic platform for both students and academic stuff for studying , learning and communicating.

Badkan auto code tester is a free platform for academic stuff to apply class assignments to. It receive a Github repository link and returns grade and feedback to the user.

Also, its can save the grade history of the student, which helps to the academic stuff to see his student progress.

Our main goal is to enable Moodle academic users to submit an academic code assignments with an automatic interface testers designed by the academic stuff and create a plug-in that will enable to connect between the "Moodle" system and "Badkan" website which is an automatic tester.

1.2 Project Scope

- the plug-in will be design to work with an automatic interface tester "badkan" website
- the plug-in may support connection with other systems such the "Badkan" in this project scope.
- Our plugin will connect the student to the "Badkan" code runner. As a result, in case of mistakes and issues with the student code, the student will get a chance to current its mistakes and learn better. There for, an academic institutions will have the option to change there teaching strategy.

1.3 References

For the Moodle website visit: https://docs.moodle.org/310/en/Main_page

For the Badkan code tester website visit: <http://bad-kan.com>

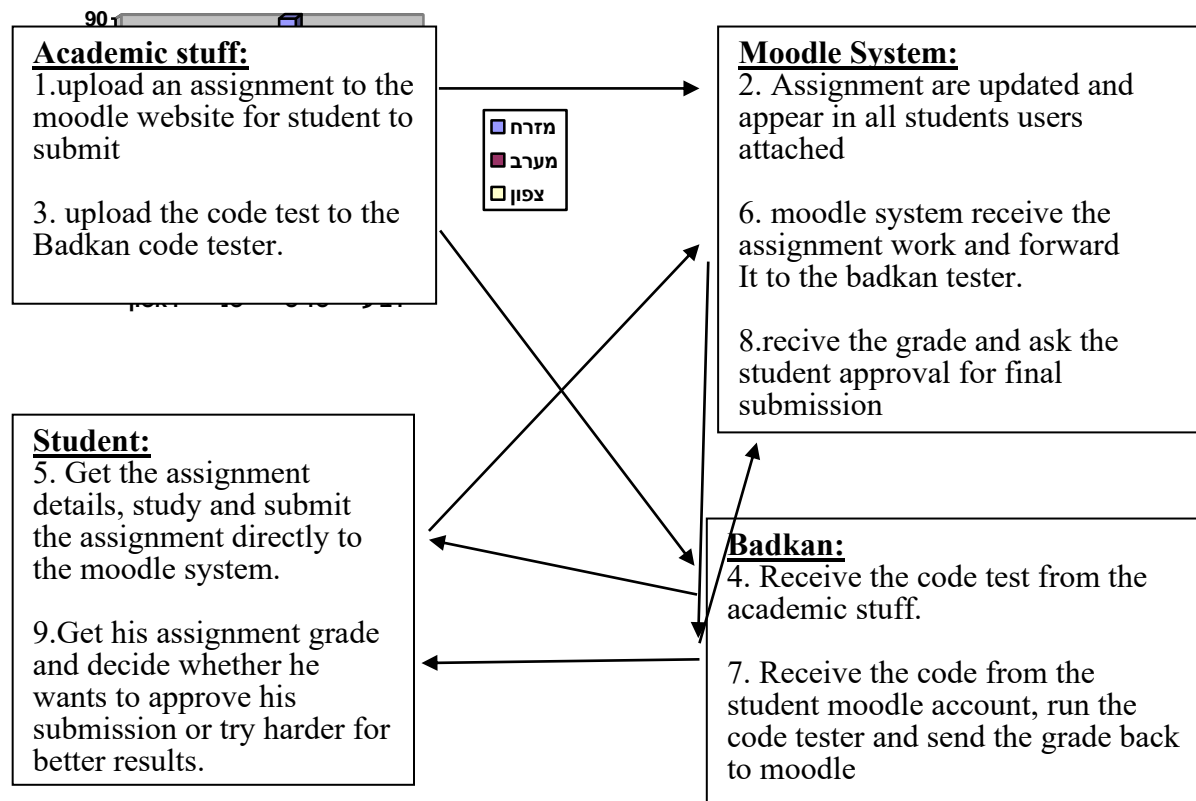
2. Overall Description

2.1 Product Perspective

Our software is a plug-in feature installed on the Moodle software system.

As usually, academic stuff upload an assignment to the moodle website for student to submit, then, the student submit the assignment directly to the moodle system and automatically get the his submission grade from the badkan code tester and decide whether he wants to approve his submission or try harder for better results.

2.2 Product Features



2.3 Operating Environment

Our product will be promoted in the open-source environment of moodle platform.

Most of the operating will be written with PHP Development language, and java script.

The plugin will work in the model system.

2.4 Design and Implementation Constraints

Our product implementation at the public distribution of the moodle system is depended on the moodle company approval to add our plug- in to there open-source code.

The system must have an internet connection to work properly.

The system should secure the login details of the academic stuff to the moodle so it can add the grade throw his account to the student moodle.

3. System Features and Functional requirements

Submission Box

new and separate submission button to submit the code to.

Priority-5

Functional Requirements:

1.The Submission Box will enable every user to enter a "GitHub" repository link of his assignment code and deliver it to the Badkan grading system.

Priority-5

2.In case of Broken link\not allowed link, user will get an error message elaborating and explaining the error source\nreason and an help with solving it.

Priority-5

3. The academic stuff will have to assign a unique Badkan-tester (Url) to the moodle assignment.

Priority-5

Feedback window

Grade feedback available for both the student and the academic stuff .

Functional Requirements

1.the grade and feedback will be available via the moodle (it will be saved in a table in moodle).

Priority-5

2. There will be a separated window opened up which will contain the grade from the Badkan code tester.

Priority-5

3 . The academic stuff will have the ability to edit this (1) window.

Priority-2

4. The window will contain the full feedback from the Badkan code tester.

Priority-5

5 . in cases the code wouldn't compile due to code error/structure error.

Priority-5

4. Other Nonfunctional Requirements

4.1 Performance Requirements

1. Grades will be presented in the user window in a few second later the badkan tester will finish its process.
Priority-3
2. The feedback will be presented in a nice and clear way, emphasizing the relevant lines.
Priority-2
3. *The plug-in will be easy to install and easy to use.*
Priority-4

4.2 Security Requirements

The user information will be accessible only to him and to the academic stuff.

4.3 Software Quality Attributes

The plug-in will be an open-source project which will allows other developers to add/contribute to – Allow more code Testers to connect it with.