Initial Plan

Bungo: A spaced-repetition based code-kata platform.

Author: Thomas Mahony-Kelross C1714148 Supervisor: Martin Chorley Module: CM3203 (One Semester Individual Project) - 40 Credits

Project Description

Context

In Karate, a kata is a "combination of positions and movements" [0] which are repeated, over and over again, in order to perfect the form of the movements.

A code kata (a term coined by Dave Thomas[1]) re-applies this concept to the domain of software development. It is a programming exercise, designed to help anyone who writes code improve their skills through repeated practice.

A code kata typically consists of three parts; a prompt, input from the user in the form of code and a hidden set of test cases used to evaluate the correctness of the user's input.

What is the optimal rate to repeat these exercises?

The spacing effect[2] is the phenomena that distributing learning over numerous small separate learning sessions leads to better long-term results than one large continuous session (cramming).

Spaced repetition[3], typically used with flash-cards in secondary language acquisition, is the practice of algorithmically determining when these learning sessions should take place to make most use of the spacing effect.

The problem and aims

The overarching aim of this project is to create a platform which will enable the authoring and (spaced repetition based) practice of code katas, with a focus on increasing the effectiveness of recent or to-be Cardiff computer science graduates in the Python programming language.

Project aims and objectives

Scope constraints

- From here on out the term kata refers to a code kata.
- The project will focus primarily on katas for the Python programming language.
- The project will focus on three stakeholders / types of users.
 - Creators:
 - General motivation is to be able to create katas and share them with Learners.
 - Learners:
 - General motivation is to be able to practice katas and improve their proficiency in them over time.
 - Administrators
 - General motivation is to provide and maintain the platform for Creators and Learners.

Core aims

Aim: Enable Creators to author a kata. **Objectives:**

- Determine a serialisation format(s) for a kata, including its prompt and test cases.
- Demonstrate an example workflow that would allow katas to be created in this format.
- Optionally: Explore how this workflow could be made more effective and accessible for Creators in the future.

Aim: Enable Learners to practice a kata.

Objectives:

(Understanding practice to mean providing input in the form of code, in response to a kata's prompt)

- Create a mechanism to determine the correctness of a Learner's input for a kata based on that kata's test cases.
- Create a web-application for Learners to practice through.
- Optionally: Provide alternative interfaces for Learners to practice though

Aim: Use spaced repetition to determine when a Learner will next be prompted to practice a kata.

Objectives:

- Implement a well known spaced repetition algorithm (such as the SuperMemo 2 algorithm)
- Optionally: Enable Learners to specify their own spaced repetition algorithm.

Aim: Store authored katas, the results of Learners practice and any other critical data. **Objectives:**

- Design an appropriate data model.
- Implement this data model using a DBMS technology.
- Create an interface to this data store.

Secondary Aims

These Aims represent ways of extending the project past its core functionality. It is anticipated that it may only be feasible to realise one or two of these goals within the time constraints of the project. Multiple aims are provided here to provide different paths for the project to continue once its core functionality has been implemented.

Aim: Ensure the platform has a robust security model.

Objectives:

- Document the platforms security boundaries
- Document the platform's actors and levels of trust between them.
- Implement appropriate authentication for the platforms infrastructure and services.
- Implement appropriate sandboxing mechanisms to safely execute trusted and untrusted code.

Aim: Where possible, Learners should be able to practice their kata without an internet connection.

- Determine a reasonable set of constraints under which offline functionality could be provided
- Attempt to implement this offline functionality.
- Explore what a reasonable synchronisation method for when the connection is restored could look like.

Aim: It should be possible to provide alternative clients in the future.

- Explore how the platform would work in a 'headless' capacity, eg without an accompanying web-application
- Explore how this 'headless' capacity could be used to provide alternative clients such as mobile applications.

Aim: Ensure portability of data between different instances of the platform.

- Enable katas authored on a different instance of the platform to run properly with minimal configuration
- Enable all users to import and export their data from different instances of the platform
- Explore potential ways to import and export data to other popular software systems
- Explore potential uses of Open Linked data and federated technology.

Work Plan

Week	Objectives	
1	 Initial plan Domain research Decide on tooling and infrastructure. Seek ethical approval for user testing where required. 	
2	 Design state machine for a Kata lifecycle Implement SuperMemo2 spaced repetition algorithm Implement a 'Hello world' Kata 	
3	 Implement REST API for data storage Implement REST API for Kata execution Create 10 example python Kata's 	
4	 Focus on front-end Use Data API to retrieve Kata Use Execution API to practice a Kata Use spaced-repetition to determine when next practice should occur Persists practice results with Data API 	
Milestone: Minimal Viable Product A software system that satisfies all of the 'core aims', allowing users to author and practice katas in a spaced repetition based way.		
5	Choose three of the four following sprints, dedicating roughly two weeks to each and allowing two weeks of slack for any difficulties, and any other family or academic	
6	commitments during the easter recess.	
7	For each sprint the time will be spent attempting to realise the objectives of the relevant secondary aim.	
8	Security Sprint	
9	Offline Sprint	
10	Alternative Client Sprint	
11	Federation Sprint	
12	User testing should also occur during this period.	
Milestone: Expanded Product A software system that satisfies all of the 'core aims' and at least one of the secondary aims.		

13	 Code freeze apart from bug fixing, testing and documentation. Conclude user testing and analyze results. Write and publish the final report.
14	
15	

Review meetings will occur on a fortnightly basis every tuesday afternoon.

Ethics

I would like to conduct user testing in this project.

Primarily I would like to conduct a qualitative survey, measuring users perceived competence at python programming before and after using the project for a number of weeks.

If possible I would also like to extend the platform to collect quantitative measures of users performance.

To ensure I comply with the School's ethics procedure I will the complete the Research Integrity Online Training Programme by the end of the third week and I will send off an ethics approval form by the end of week six at the latest.

[0] https://web.archive.org/web/20191108092845/https://www.merriam-webster.com/dictionary/kata

[1] <u>https://web.archive.org/web/20140420033915/http://codekata.com/kata/codekata-intro/</u> and Hunt, A., Thomas,

D.,, Cunningham, W. (1999). The Pragmatic Programmer. From Journeyman to Master. Addison-Wesley Longman, Amsterdam. ISBN: 020161622X

[2] https://web.archive.org/web/20191108092845/https://www.merriam-webster.com/dictionary/kata

[3] https://web.archive.org/web/20210131022230/https://en.wikipedia.org/wiki/Spaced_repetition