



## **CoderDojo Community Platform**

Platform encompassing all web services for  
the CoderDojo Community

### **Requirements Specification**

*Date :* 10th December 2014

*Authors:* Laura Ivers  
Eugene McDonough

*Version:* 3

*Status:* Draft for Review

## Document Control

Contributors		
<b>Name/Position</b>	<b>Organization</b>	<b>Contact Details</b>
Laura Ivers	CoderDojo Foundation	laura@coderdojo.org
Eugene McDonough	CoderDojo Foundation	eugene@coderdojo.org

Version History			
<b>Date</b>	<b>Version</b>	<b>Status</b>	<b>Comments</b>
	3	For Review, shared externally.	

<b>Changes since last version</b>

Known Omissions

## Reviewers Comments

No comments

# *Table of Contents*

## [1 Introduction](#)

[1.1. Purpose of the Document](#)

[1.2 Intended Audience](#)

[1.3 Scope of solution](#)

[1.4 User Types Overview](#)

[1.5 Definitions, Acronyms and Abbreviations](#)

## [2 Requirements Overview](#)

[2.1 Functional Scope](#)

[2.2 Initial Specification](#)

[2.3 Initial Requirements List](#)

## [3 Requirements Definition](#)

[3.1 High Level overview](#)

[3.2 Functional Modules](#)

[3.3 Requirements List](#)

[FM-0001](#) [Registration/Profile Creation](#)

[3.3.1 FM-0002](#) [Dojo Listing](#)

[3.3.2 FM-0003](#) [Dojo Email](#)

[3.3.3. FM-0004](#) [Dojo Ticketing](#)

[3.3.4 FM-0005](#) [Dojo Adult Forum](#)

[3.3.6 FM-0006](#) [Dojo Youth Forum](#)

[3.3.7 FM-0007](#) [Dojo IM](#)

## [4 Technical Specification](#)

## [5 Sitemap](#)

## [6 References](#)

# 1 Introduction

## 1.1 Purpose of the Document

The purpose of this document is to record the requirements for the *CoderDojo Community Platform* which will serve as an all encompassing online system for serving the technical requirements of the CoderDojo community .

The community platform is intended to help automate the tasks of Champions (such as issuing tickets, managing waiting lists and managing volunteers and attendees), to give recognition to Champions, mentors and attendees through profile pages and awarding of Mozilla Open Badges as well as giving parents and youths a method of booking and viewing tickets for their Dojo. The platform will also help feedback high level statistics and reporting to the CoderDojo Foundation on the CoderDojo Community.

## 1.2 Intended Audience

This document is intended as an overview of the proposed requirements for the CoderDojo Community Platform. The intended audience is potential stakeholders in the development of the project.

## 1.3 Scope of Project

- We envision this part of project being completed in 3 phases or iterations with the final iteration and testing to be complete by the 1st of September, 2015. After this there may be other functionality to add in but for now the project is to be broken into 3 phases as detail in the Initial High Level Requirements List at 2.3.
- This project must align with the CoderDojo ethos, it is a requirement that the company who develop the CoderDojo Community platform should have a proven track record contributing to open source projects and that open source components are used wherever practical.
- It is a requirement that the software is well documented and modularized to make it easier for the community to make updates and additions to later versions. Must also be hosted on Github or similar to ensure the community can access and request additions.
- It is a requirement that software created for CoderDojo Foundation is owned by the CoderDojo Foundation to license according to their policies or licensed to CoderDojo Foundation under an agreed open source license. We intend to share the code for this platform for the open source community and other organisations to use as they please.
- It is a requirement that this platform is mobile compatible with the long term view of developing a mobile app.
- The company who engage is this project must be open to hearing the suggestions, thoughts and feedback of CoderDojo community members. Our community has a wealth of technical expertise and we would like this to have an impact on the

Community Platform.

- We would require extensive user testing on this project prior to deployment.
- COPPA guidelines must be adhered to for all users under 13 years of age. Please see <http://www.coppa.org/> for more information.

## 1.4 User Types Overview

### 1.4.1 **Champion**

CoderDojo local organiser. In charge of organising Dojo dates, mentors, content covered etc. Needs to have a login and profile, Dojo listing creation functionality, access to Dojo email, ticket issuing and waiting list functionality, adult forum, instant messaging. Also to have control to approve platform registrations and permissions for other users from their Dojo, eg. can grant admin permissions to mentors and parents. Champions who have proven background checking can have access to youth forum.

### 1.4.2 **Mentor**

CoderDojo volunteer technical mentor. Responsible for coming to a Dojo and helping attendees to learn with technology. Need access to login, profile. ticket booking and viewing and access to adult forum. Can be granted Admin permissions by Champion to assist with ticketing etc.

### 1.4.3 **Attendee/Youth Over 13**

CoderDojo attendee. Comes to a Dojo to learn technology. Needs access to sign on, profile,, ticket booking and viewing, access to youth forum.

### 1.4.4 **Attendee/Youth Under 13**

CoderDojo attendee. Comes to a Dojo to learn technology. Parent/Guardian creates a subaccount for them,

### 1.4.5 **Parent/Guardian**

Parent of attendee (often mentor also). Needs access to sign on, ticket booking and viewing. Can be given Dojo Admin permissions to help with ticket issuing etc.

### 1.4.6 **CoderDojo Foundation Staff**

CoderDojo staff who will use the platform to communicate and serve the CoderDojo community. Needs to have varying levels of access to the platform, including edit control over all accounts, approval of account registrations etc.

## 1.5 Definitions, Acronyms and Abbreviations

**CoderDojo:** Global 550+ community of CoderDojo free programming clubs for young people. See <http://coderdojo.com/>

**CDF/CoderDojo Foundation/Foundation:** Small team based in Dublin who support and

scale CoderDojo. See <http://coderdojo.org/>

**CDF Admin:** A member of Foundation staff who manages the platform.

**COPPA:** Children's Online Privacy Protection Act

**Dojo:** A singular CoderDojo club.

**Champion:** Local volunteer CoderDojo Organiser

**Mentor:** Technically skilled volunteer who shares their knowledge at a Dojo

**Mozilla Open Badges:** Displayable digital badges to certify digital skills. Please see: <http://openbadges.org/>

**Youths/Ninjas/Attendees:** Young person who attends a Dojo.

**Onboarding/Registration:** After registration users are onboarded into a role eg. Champion, Mentor, Parent, Youth under 13, youth over 13.

**Zen:** CoderDojo database of Dojo, existing version viewable at <http://zen.coderdojo.com/>

**Kata:** CoderDojo community wiki, MediaWiki, viewable at <http://kata.coderdojo.com/>

**Organisers Google Group:** Google group for Dojo Champions and mentors to discuss important topics and issues viewable here: <https://groups.google.com/forum/#!forum/coderdojo-org>

**IM:** Instant Messaging for logged in users (adults only)

# 2 Requirements Overview

## 2.1 Functional Scope

Provision of an all encompassing platform that will enable CoderDojo community members to complete all of their Dojo management tasks in one place.

## 2.2 Initial Specification

The following specification is being given at the project inception:

1. Platform to be an amalgamation of CoderDojo email function, Organisers Google Group and Zen listing functions into one platform with single sign on.
2. Addition of user profiles for Champions, Mentors and Youths and accounts for parents and their U13 youths, ticket booking and waiting list functionality, group forums, instant messaging support functionality for CoderDojo Foundation staff.

## 2.3 Initial High Level Requirements List

Phase	Description
1	Zen reproduction
1	Improved Champion onboarding process eg. UI walkthrough, display percentage complete etc.
1	Champion registration and profile creation
1	Champion can create, edit and delete multiple Dojo listings. Champion can invite a co-Champion.
1	Transfer of all existing Zen accounts to new Zen platform
2	Single sign on for all CoderDojo sites, including Kata (media wiki), OAuth
2	Champion can add Mentors as members of their Dojos or users can request to join a Dojo as a mentor and be approved by Champion.
2	Mentor profile creation, access to adult forum, viewing and booking tickets.
2	Dojo Admin permissions enabled for Champion to grant users certain permissions to access ticket issuing and Dojo mail.
2	Dojo Mail, management of Dojo email, Champion and permitted Dojo Admin have access.
2	Adult forum, publicly viewable forum for Champions and Mentors. Users must have Mentor permissions to access.

3	Youth (Over 13) create profile and has access to Dojo tickets, youth forum. Over 13 Youth accounts can be given Mentors permissions and Dojo Admin permissions by Champion.
3	Parent/Guardian Accounts: Parents/Guardians create an account, have login and password and access to Dojo tickets. Can add a sub account for U13 youth users.
3	Under 13 Youth Sub Accounts: Added by parent/guardian, can access ticket viewing, Mozilla open badges, can create private profile.
3	Ticketing for Dojos, Issuing tickets, to include waiting list functionality (Accessed by Champions and those with approved Dojo Admin permissions).
3	Youth Forum: Over 13 users can access CoderDojo Youth chat forum to share discussions, Mentors and Champions who have proven they are background checked can access youth forum.
3	Integration of Mozilla Open Badges, can be earned and displayed through Youth profiles.
<b>All Phases</b>	
All	CRM Connecting - All phases
All	3rd Party Reporting - All phases
All	Logging - All phases



# 3 Requirements Definition

## 3.1 High Level Overview

Based on the requirements gathering and analysis phases described above the following high level model of the system has been derived.

## 3.2 Functional Modules

The functional modules describe the main functional clusters within the system.

Ref	Name	Description	Phase
FM-01	Zen Reproduction	Reproduce the existing functionality of Zen using the new architecture.	1
FM-02	Champion onboarding/Profile Creation/Champion Registration	Potential CoderDojo Champion initial registration, then onboarding through to actual Champion with Dojo listing with assistance from CoderDojo Foundation staff. eg. UI walkthrough, display percentage complete etc. Champion can invite a co-champion.	1
FM-03	Dojo Listing	Registered Champion can create additional Dojos listings and edit existing publicly viewable Dojo listing/profile	1
FM-04	OAuth	Single sign on for all users across all CD web services including Kata and Zen/Platform.	2
FM-05	Mentor Profiles	Dojo Mentors can be invited to join the platform by their Dojo Champion or request to join a Dojo and once approved can create profiles and access adult forum and Dojo ticket viewing and booking.	2
FM-06	Dojo Admin	Mentors (and other users linked to that Dojo) can be granted Dojo Admin permissions by Champion to access Dojo listing, ticket management, email etc.	2
FM-07	Dojo Mail	Integration of management of current CoderDojo email (through Gmail) for the Champion of the Dojo.	2
FM-08	Adult Forum	Publicly viewable forum for Champions and mentors to discuss CoderDojo related topics. User must have Champion or Mentor privileges to access.	2

FM-09	Youth Profile (Over 13)	Youths are invited/can request to create profile (optionally publicly viewable) in connection with a Dojo Champion, have access to Dojo tickets and youth forum. Can complete Mozilla open badges through platform and can display on profile. Can be granted Mentor and Dojo Admin permissions.	3
FM-10	Parent/Guardian Profile	Parent can create an account with login (no profile) and create profile for Under 13 Youths. Can be granted Mentor/Dojo Admin permissions and upgraded to having a Mentor profile. Can reset Youth (under 13) sub account password.	3
FM-11	Youth Profile (Under 13)	Youth sub user account added by parent/guardian account, private profile, can access ticket viewing (not booking) and Mozilla open badges. Can choose unique username and password to manage their private profile. Parent account books tickets.	3
FM-12	Dojo Ticketing	Ticketing for Dojos, Issuing tickets, to include waiting list functionality. Accessed by Champions and those with approved Dojo Admin permissions. Check in at Dojo	3
FM-13	Youth Forum	Over 13 users can access CoderDojo Youth chat forum to share discussions, Mentors and Champions who have proven they are background checked can access youth forum.	3
FM-14	Mozilla Open Badges Integration	Youth users can complete Mozilla Open Badges through platform and display completed badges on their profile.	3
All	CRM	CRM Connecting - All phases	All
All	Reporting	3rd Party Reporting - All phases	All
All	Logging	Logging (Log files) - All phases	All

### 3.3 Requirements List

#### 3.3.1 FM-01 Zen Reproduction

Reproduce the existing functionality of Zen using the new architecture.

Ref	Description
Public	
FR - 1.010	Public user views map of Dojos
FR - 1.020	Public user searches map Dojo using location and is shown list of nearest Dojos.
FR - 1.030	Public user browses Dojo listing by country and region.
FR - 1.040	Public users can click on link to and view Dojo profile.
Champion	
FR - 1.050	Champion registration.
FR - 1.060	Champion login
FR - 1.070	Champion sign CoderDojo Charter (Obligatory to create account)
FR - 1.080	Champion creates listing for Dojo
FR - 1.090	Champion can set the status of their dojo.
FR - 1.100	Example fields: Dojo Status, Location, GPS shown on map, display ticketing for Dojo, contact details
CDF Admin	
FR - 1.110	CDF Staff to review and verify Dojo listings.
FR - 1.120	CDF Staff to have full access to all profiles.
FR - 1.130	CDF Staff to have full access and moderation access rights on Adult and Youth Forum
FR - 1.140	Autopost status update to Twitter/Facebook when a new Dojo listing is verified.

### 3.3.2 FM-02 Registration/Profile Creation/Onboarding

Potential CoderDojo Champion initial registration, then onboarding through to actual Champion with Dojo listing with assistance from CoderDojo Foundation staff. eg. UI walkthrough, display percentage complete etc. Champion can invite a co-champion.

Ref	Description
Public	
FR-2.010	New visitor creates login.
FR-2.020	User logs in.
Champion	
FR-2.030	User requests to create a new dojo or join an existing Dojo as champion
FR-2.040	Champion onboarding process provides all information required for champion profile as well as extra private information for CDF Admin only.
FR-2.050	System creates champion profile.
FR-2.060	User can edit and delete profile and option to mark public/private (Champions must be public)
FR-2.070	Champion must first create profile for themselves, then Dojo listing then approve all other accounts that register for that Dojo.
FR-2.080	User can view other profiles from their Dojo
FR-2.090	Background checking validation process for Champions.
FR-2.100	Background checked Champions can access youth forum.
FR-2.110	Champions can invite Dojo members to join platform and create account.
FR-2.120	Champions have access to adult forum.
FR-2.130	After x months of inactivity users are notified that their account will become inactive after X days. Account is added to list to be reviewed by champion and champion is emailed. If no action the account is marked as inactive.
CRM	<ol style="list-style-type: none"> <li>1. "Account" is created and converted based on the relevant onboarding process.</li> <li>2. All updates to account information are passed to CRM.</li> <li>3. All updates to account status are passed to CRM</li> <li>4. All future updates will also update the CRM</li> </ol>
Reporting	All status changes to users will initiate an API call for reporting purposes.
Logging	All changes to a user status or profiles will be logged with date/time and user with details of the changes.

### 3.3.3 FM-03 Dojo Listing

Allow Champion and approved Mentors to create and edit a Dojo Listing/Profile.

Ref	Description
FR-3.010	Logged in Champion can create a listing/profile for their Dojo.
FR-3.020	Example fields: Location, GPS shown on map, display ticketing for Dojo, contact details
FR-3.030	All listings viewable on searchable public map (no login required)
FR-3.040	Potential/new champion has a Zen listing first, then has to add themselves as a member of that Dojo
FR-3.050	CDF Staff to review and verify Dojo listings.
FR-3.060	Autopost status update to Twitter/Facebook when a new Dojo listing is verified.
CRM	<ol style="list-style-type: none"> <li>1. A new account will be created for the Dojo.</li> <li>2. The "Account" for the user will be converted to champion and added under the Dojo.</li> <li>3. As each step on the onboarding process is completed it will be updated in the CRM.</li> <li>4. All future updates will also update the CRM</li> </ol>
Reporting (	All status changes to the Dojo will initiate an API call for reporting purposes.
Logging	All changes to a listing will be logged with date/time and user with details of the changes.

### 3.3.4 FM-04 OAuth

Ref	Description
FR-4.010	Platform will be a Oauth Authorization Server.
	CDF admin can a new service for a resource server with all the required attributes ie. Kata, forum
	User can connect from their client to the resource server and request access to it using their platform credentials.
	Resource server request approval from Authorization Server / user must log into Authorization Server or be logged in already.
	Authorization Server returns approval and the date it is approved to share with the resource server ie Access level = Mentor
	User can now access the resource server with the same credentials as the platform for

	as long as they credentials are alive.
--	--

### 3.3.5 FM-05 Mentor Profiles

Dojo Mentors can be invited to join the platform by their Dojo Champion or request to join a Dojo and once approved can create profiles and access adult forum and Dojo ticket viewing and booking.

Ref	Description
FR-5.010	Mentor can be invited to join a Dojo by Champion
FR-5.020	Mentor can request to join a Dojo and be approved by Champion
FR-5.030	Mentor has public profile
FR-5.040	Mentor has access to ticket viewing and booking
FR-5.050	Mentors has access to Adult Forum
FR-5.060	Mentor can be granted Dojo Admin permissions
FR-5.070	Background checking validation process for mentors.
FR-5.080	Background checked mentors can access youth forum.
FR-5.090	Mentor can join multiple Dojos.
CRM	<ol style="list-style-type: none"> <li>1. User registration creates account on Zen</li> <li>2. As each step on the onboarding process is completed it will be updated in the CRM.</li> <li>3. The "Account" for the user will be converted to Mentor and added as under the Dojo</li> <li>4. All future updates will also update the CRM</li> </ol>
Reporting	All status changes to the Mentor status will initiate an API call for reporting purposes.
Logging	All changes to a listing will be logged with date/time and user with details of the changes.

### 3.3.6 FM-06 Dojo Admin

Non Champion users linked to a Dojo can be granted Dojo Admin permissions by Champion to access Dojo listing, ticket management, email etc.

Ref	Description
-----	-------------

FR-6.010	Champion can grant users (Mentor, Parent, Over 13 Youth) Dojo Admin permissions.
FR-6.020	Can be granted standard permissions (eg. Access to edit Dojo listing, access to add mentors and approve parents and youth members and access to ticket issuing and management) or just choose specific permissions for user.
CRM	<ol style="list-style-type: none"> <li>1. The “ existing Account” will be flagged as an admin.</li> <li>2. All future updates will also update the CRM</li> </ol>
Reporting	All status changes to the user status will initiate an API call for reporting purposes.
Logging	All changes to a listing will be logged with date/time and user with details of the changes.

### 3.3.7 FM-07 Dojo Mail

Integration of management of current CoderDojo email (through Gmail) for the Champion of the Dojo.

Note: On verification each Dojo is issued a @coderdojo.com email address. We would like to investigate if the creating of this email address can be done through the Google Apps API and if not we would like to put a management system in place to track what address need to be created and made inactive.

Ref	Description
FR-7.010	Each Dojo gets it's own email address in the format [DOJONAME].[COUNTRYISO]@coderdojo.com eg. <a href="mailto:dublin.ie@coderdojo.com">dublin.ie@coderdojo.com</a>
FR-7.020	The email address is created in Google apps using the API or added to admin list to be manually created and marked as done.
FR-7.030	If a Dojo become inactive the email address should marked as inactive by API call or added at admin list to be manually made inactive and marked as done.
FR-7.040	Email address will be automatically added to profile when marked as done.
FR-7.050	Email activity logged and reported.
CRM	Email address will be added to the Dojo account.
Reporting	No update needed.
Logging	All changes will be logged with date/time and user with details of the changes.

### 3.3.8 FM-08 Adult Forum

Publicly viewable forum for Champions and mentors to discuss CoderDojo related topics. User must have Champion or Mentor privileges to access.

Ref	Description
FR-8.010	Users can access forum and view all existing threads/conversations
FR-8.020	Public (non logged in) can view adult forum and conversations.
FR-8.030	Users can comment on existing threads
FR-8.040	Users can start a new thread
FR-8.050	CoderDojo Foundation users can remove inappropriate threads.
CRM	No update needed.
Reporting	No update needed.
Logging	No update needed.

### 3.3.9 FM-09 Youth Profile (Over 13)

Youths are invited/can request to create profile (optionally publicly viewable) in connection with a Dojo Champion, have access to Dojo tickets and youth forum. Can complete Mozilla open badges through platform and can display on profile. Can be granted Mentor and Dojo Admin permissions.

Ref	Description
FR-9.010	Youth can be invited to join the Dojo
FR-9.020	Youth can request to join a Dojo
FR-9.030	Youth can create a profile (Optionally public/private)
FR-9.040	Youth has access to ticketing viewing and booking.



FR-9.050	Youth has access to youth forum.
FR-9.060	Youth can be granted user permissions.
FR-9.070	Youth can join multiple Dojos.
FR-9.080	Youth can complete Mozilla Open Badges through profile.
FR-9.090	Youth can display completed Badges on their profile.
CRM	<ol style="list-style-type: none"> <li>1. User registration creates account on Zen</li> <li>2. The "Account" for the user will be converted to Youth and added as under the Dojo</li> <li>3. All future updates will also update the CRM</li> </ol>
Reporting	All status changes to the Youth status will initiate an API call for reporting purposes.
Logging	All changes to profile will be logged with date/time and user with details of the changes.

### 3.3.10 FM-10 Parent/Guardian Profile

Parent can create an account with login (no profile) and create profile for Under 13 Youths. Can be granted Mentor/Dojo Admin permissions and upgraded to having a Mentor profile. Can reset Youth (under 13) sub account password. Please see Khan Academy for example of parent profile,

Ref	Description
FR-10.010	Parent can be invited to join a Dojo.
FR-10.020	Parent can request to join a Dojo.
FR-10.030	Parent has private profile
FR-10.040	Parent has access to ticket booking and viewing.
FR-10.050	Parent can add an account for under 13 user through their account.
FR-10.060	Parents can be granted Mentor and Dojo Admin permissions.

CRM	<ol style="list-style-type: none"> <li>1. User registration creates account on Zen</li> <li>2. The "Account" for the user will be converted to Parent and added as under the Dojo</li> <li>3. All Youths under 13 added by a parent will be note on CRM</li> <li>4. All future updates will also update the CRM</li> </ol>
Reporting	All status changes to the parent status will initiate an API call to SAS for reporting purposes. Numbers of youths under 13 will be passed for reporting.
Logging	All changes to profile will be logged with date/time and user with details of the changes.

### 3.3.11 Youth Profile (Under 13)

Youth sub user account added by parent/guardian account, private profile, can access ticket viewing (not booking) and Mozilla open badges. Can choose unique username and password to manage their private profile. Parent account books tickets.

Ref	Description
FR-11.010	Under 13 Youth user is added by registered and approved parent.
FR-11.020	Under 13 Youth user can create a private profile, viewable only by themselves and their guardian.
FR-11.030	Under 13 Youth user has access ticket viewing (not booking)
FR-11.040	Can complete Mozilla Open Badges
FR-11.050	Can display completed Badges on private profile.
FR-11.060	Is prompted to upgrade to an Over 13 profile when of age.
CRM	On conversion to over 13 account: <ol style="list-style-type: none"> <li>1. New Youth account created under the Dojo.</li> <li>2. All future updates will also update the CRM</li> </ol>
Reporting	All status changes to the Youth status will initiate an API call for reporting purposes.
Logging	All changes to profile will be logged with date/time and user with details of the changes.

### 3.3.12 Dojo Ticketing

Ticketing for Dojos, Issuing tickets, to include waiting list functionality. Accessed by Champions and those with approved Dojo Admin permissions. Check in at Dojo

<b>Ref</b>	<b>Description</b>
FR-12.010	Champions and users with Dojo Admin permissions can schedule and event and create an event profile.
FR-12.020	Approved users can set an event to repeat on a regular basis.
FR-12.030	Approved users can release a certain amount of tickets for their event.
FR-12.040	Different ticket types, eg. standard (book each week), term ticket (eg. 6 weeks), other/customisable.
FR-12.050	Approved users can access waiting list for extra bookings.
FR-12.060	Waiting list synced to ticketing for automatic issuing of tickets when a ticket is cancelled.
FR-12.070	All users, and public, can view events and book tickets.
FR-12.080	All users, and public , can view and edit their booked tickets.
FR-12.090	Public users create minimum login, no profile, in order to book tickets. (Mostly youth and parents)
	Open badge can be linked to event and automatically awarded to all checked in attendees.
Reporting	Event completion will be reported. No of attendees, ages and gender will be reported.
Logging	All changes to the event will be logged with date/time and user with details of the changes.

### 3.3.13 Youth Forum

Over 13 users can access CoderDojo Youth chat forum to share discussions, Mentors and Champions who have proven they are background checked can access youth forum.

<b>Ref</b>	<b>Description</b>
FR-13.010	Youth users can access forum and view all existing threads/conversations

FR-6.020	Users can comment on existing threads
FR-6.030	Users can start a new thread
FR-6.040	CoderDojo Foundation users can remove inappropriate threads.
FR-6.050	Approved and background checked mentors and Champions can access youth forum.
CRM	No update
Reporting	No update
Logging	No update

### 3.3.14 Mozilla Open Badges Integration

Youth users can complete Mozilla Open Badges through platform and display completed badges on their profile.

Ref	Description
FR - 14.010	Users can be linked to complete a Mozilla Open Badge through their profile.
FR - 14.020	Mentors/Champions can approve/verify that young people have completed their badges.
FR -14.030	Users will be able to display Mozilla Open Badges on their profile pages.
FR-14.040	U13 users complete their badges through their parents.
FR-14.050	Functionality to export CoderDojo Badges to Mozilla Backpack.
FR-14.060	Event based badges - attended 100 event and special events
CRM	Update details of badges to profile
Reporting	Report on badges issued.

Logging	All changes to badges will be logged with date/time and user with details of the changes.
---------	---

## 4 Technical Specification

Product Name:	CoderDojo Community Platform																				
Product Type:	Web Platform																				
Application Type:	Community Management System																				
Delivery Mechanism	Hosted on a dedicated server Zen - <a href="http://zen.coderdojo.com/">http://zen.coderdojo.com/</a> Froum - <a href="http://froum.coderdojo.com">froum.coderdojo.com</a>																				
Target Device:	<table border="1"> <thead> <tr> <th>Browser OS:</th> <th>Mobile Devices:</th> </tr> </thead> <tbody> <tr> <td>1 Chrome</td> <td>1 iPhone</td> </tr> <tr> <td>2 Safari</td> <td>2 iPad</td> </tr> <tr> <td>3 FireFox</td> <td>3 Samsung Galaxy S IV</td> </tr> <tr> <td>4 Internet Explorer</td> <td>4 Google Nexus 5</td> </tr> <tr> <td>5 Android Browser</td> <td>5 Google Nexus 7</td> </tr> <tr> <td>6 Safari (in-app)</td> <td>6 HTC One</td> </tr> <tr> <td>7 Opera</td> <td>7 Samsung Galaxy S V</td> </tr> <tr> <td>8 Amazon Silk</td> <td>8 Samsung Galaxy S4 mini</td> </tr> <tr> <td>9 IE with Chrome Frame</td> <td>9 Samsung Galaxy S III</td> </tr> </tbody> </table>	Browser OS:	Mobile Devices:	1 Chrome	1 iPhone	2 Safari	2 iPad	3 FireFox	3 Samsung Galaxy S IV	4 Internet Explorer	4 Google Nexus 5	5 Android Browser	5 Google Nexus 7	6 Safari (in-app)	6 HTC One	7 Opera	7 Samsung Galaxy S V	8 Amazon Silk	8 Samsung Galaxy S4 mini	9 IE with Chrome Frame	9 Samsung Galaxy S III
Browser OS:	Mobile Devices:																				
1 Chrome	1 iPhone																				
2 Safari	2 iPad																				
3 FireFox	3 Samsung Galaxy S IV																				
4 Internet Explorer	4 Google Nexus 5																				
5 Android Browser	5 Google Nexus 7																				
6 Safari (in-app)	6 HTC One																				
7 Opera	7 Samsung Galaxy S V																				
8 Amazon Silk	8 Samsung Galaxy S4 mini																				
9 IE with Chrome Frame	9 Samsung Galaxy S III																				
Target OS:	Windows/IOS/Linux/Ubuntu (Desktop) Android/Windows/Mozilla/iOS (Mobile)																				
Target Platform:	Ruby, PHP, Node JS, Python																				
Initial Language:	Localisation is a priority for global community, main languages, English, Italian, German, Spanish and French. Consider Japanese and Arabic.																				

Viewing mode	Browser window
Security	Data Protection is a priority especially with youth users.
Data	eg. Collected/stored and analysed later, define the type of database. Define the types and levels of querying that need to be available. And reports.
*Note	<p>We have been advised to use a frontend JavaScript framework such as Ember or Angular. Such frameworks use client-side data models, cache data within the browser, interact with the API only when necessary, and render views locally. This results in a user experience comparable to native desktop and mobile applications. This level of speed and responsiveness would be particularly valuable.</p> <p>It has been suggested that we use a known frontend framework for the user interface, rather than creating one from scratch. This will allow for more rapid development and prototyping, as well as giving standardised structure to the frontend codebase. This reduces the learning curve when onboarding new developers/designers.</p>

## 5 References

---

Zen, current database of Dojo listings and map of locations: <http://zen.coderdojo.com/>

Kata, community wiki: <http://kata.coderdojo.com/>

CoderDojo Organisers Google Group:

<https://groups.google.com/forum/#!forum/coderdojo-org>

Khan Academy, (Example of parents and U13 accounts): <https://www.khanacademy.org/>

Mozilla Open Badges: <http://openbadges.org/>