

Contributor Recognition Infrastructure

Hasitha Kaushan

Section 1: About You

What project are you applying for?

[Contributor Recognition Infrastructure](#)

Why are you interested in working with Oppia, and on your chosen project?

I was looking to find a way to start contributing to the open source community back in August, 2020. Unfortunately, my efforts were in vain as I could not find a proper organization where a beginner could easily get in.

In the meantime, I followed the organization list of GSoC 2020 and found Oppia among them. When I referred to the oppia repository on Github, I found it easy to get started with Oppia as everything was clearly documented for a beginner to start from scratch. It is the fundamental point that attracted me to start contributing to this wonderful community, Oppia.

I have already been working as a part of the contributor experience team since January, 2021. I chose this project since I have a good experience of working with the workflows in the contributor dashboard.

Prior experience

I have been working with Oppia since December, 2020. I have been a member of the contributor experience team since January, 2021. I worked on new features as well as bug fixing and hotfixing critical bugs. I also worked with the language accessibility team to fix critical bugs and stabilize the contributor dashboard.

- [#12540](#): This PR focused on the backend implementation for editing translation suggestions and question suggestions and the frontend implementation for editing translation suggestions. This was a highly requested feature last year.
 - [Design doc](#)
- [#12909](#): This PR focused on validating translations and disabling paragraph copy for translations.

- [#14289](#): This PR fixed a release blocking bug which prevented question submitters from submitting questions with image region interactions.
- [#14341](#): This PR fixed a critical bug in the production which prevented retrieving translations to review.
- [#13020](#): This PR fixed a flaky backend test
 - [Debugging doc](#)

All of my PR for Oppia can be found [here](#).

All of the issues that I authored can be found [here](#).

Project size

~350 hours

Project timeframe

I will be having my final exams in June. So my contributions from 1st of June - 30th June will be narrow. I would like to work in the extended timeline so that I would be able to cover the missed time period. (But if the mentor is unavailable, I would be able to work in the default time period.)

Contact info and timezone(s)

Preferred medium: *Google Chat, Google Meet, Emails*

Timezone: *IST(+5.30)*

Email: hkraclh@gmail.com

Github: [@chris7716](#)

Time commitment

In June: 10 hrs/week

From 1st of July to 12th of September: 30 hrs/week

Essential Prerequisites

Answer the following questions (for Oppia Web GSoC contributors):

- *I am able to run a single backend test target on my machine. (Show a screenshot of a successful test.)*

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL bash + - + ^ X
core.controllers.editor_test (started 21:57:15)
[datastore] Apr 18, 2022 9:57:29 PM io.gap1.emulators.grpc.GrpcServer$3 operationComplete
[datastore] INFO: Adding handler(s) to newly registered Channel.
[datastore] Apr 18, 2022 9:57:29 PM io.gap1.emulators.netty.HttpVersionRoutingHandler channelRead
[datastore] INFO: Detected HTTP/2 connection.
16:29:58 FINISHED core.controllers.editor_test: 163.1 secs
Stopping Redis Server(name="redis-server", pid=18551)...
Stopping Cloud Datastore Emulator(name="python3.7", pid=18422)...

+-----+
| SUMMARY OF TESTS |
+-----+

SUCCESS core.controllers.editor_test: 92 tests (148.4 secs)

Ran 92 tests in 1 test class.
All tests passed.

Done!
(oppia) Hasithas-Air:oppia hasithakaushan$
(oppia) Hasithas-Air:oppia hasithakaushan$
```

- I am able to run all the frontend tests at once on my machine. (Show a screenshot of a successful test.)

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL bash + - + ^ X
For more info visit https://webpack.js.org/guides/code-splitting/
@ ./node_modules/@angular/platform-browser-dynamic/fesm2015/testing.js 7:0-169 42:12-22 45:44-50 109:31-41 115:84-92 121:89-98 127:89-98
133:74-78 166:35-56 167:15-31 171:15-23 171:25-40 193:38-59 202:12-20
@ ./core/templates/combined-tests.spec.ts

WARNING in ./node_modules/@angular/core/fesm2015/core.js 29751:15-102
System.import() is deprecated and will be removed soon. Use import() instead.
For more info visit https://webpack.js.org/guides/code-splitting/
@ ./node_modules/@angular/platform-browser-dynamic/fesm2015/testing.js 7:0-169 42:12-22 45:44-50 109:31-41 115:84-92 121:89-98 127:89-98
133:74-78 166:35-56 167:15-31 171:15-23 171:25-40 193:38-59 202:12-20
@ ./core/templates/combined-tests.spec.ts
1 warning: Compiled with warnings.
18 04 2022 21:17:42.382:WARN [filelist]: Pattern "/third_party/static/lamejs-1.2.0/worker-example/worker-realtime.js" does not match any f
ile.
18 04 2022 21:17:42.383:WARN [filelist]: Pattern "/Users/hasithakaushan/Documents/opensource/oppia/extensions/interactions/**/*.*.directive.
html" does not match any file.
Chrome Headless 100.0.4896.88 (Mac OS 10.14.6): Executed 7377 of 7378 SUCCESS (6 mins 1.76 secs / 4 mins 19.862 secs)
TOTAL: 7377 SUCCESS
TOTAL: 7377 SUCCESS
18 04 2022 21:24:43.214:WARN [launcher]: ChromeHeadless was not killed in 2000 ms, sending SIGKILL.
Done!
(oppia) Hasithas-Air:oppia hasithakaushan$
```

- I am able to run one suite of e2e tests on my machine. (Show a screenshot of a successful test.)

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL
[datastore] INFO: Detected HTTP/2 connection.
.
. Feature Gating Flow
  000 should not show indicators gated by dummy feature when disabled
[datastore] Apr 18, 2022 10:39:13 PM io.gapi.emulators.grpc.GrpcServer$3 operationComplete
[datastore] INFO: Adding handler(s) to newly registered Channel.
[datastore] Apr 18, 2022 10:39:13 PM io.gapi.emulators.netty.HttpVersionRoutingHandler channelRead
[datastore] INFO: Detected HTTP/2 connection.
. 000 should show dummy feature in the features tab
. 000 should not show indicators for dummy_feature to different users

3 specs, 0 failures
Finished in 322.096 seconds

Executed 3 of 3 specs SUCCESS in 5 mins 22 secs.
[22:41:06] I/launcher - 0 instance(s) of WebDriver still running
[22:41:06] I/launcher - chrome #01 passed
Stopping Protractor Server(pid=21336)...
Stopping Webdriver manager(name="node", pid=21293)...
Nithesh N. Hariharan, 3 years ago Ln 112, Col 24 Spaces: 2 UTF-8 LF () JavaScript
```

Other summer obligations

After finishing my final exams in June, I will only focus on GSoC.

Communication channels

I am available on Google Chat and also through emails. I would be willing to have two weekly meetings with my mentor through Google Meet.

Section 2: Proposal Details



Problem Statement

Link to PRD (or N/A if there isn't one)	N/A
Target Audience	<ol style="list-style-type: none">1. Translations Submitters: They are proficient in submitting translations for cards in one or more languages.2. Translations Reviewers: They are proficient and responsible in accepting or rejecting the translations.3. Questions Submitters: They are proficient in submitting practice questions for skills.4. Questions Reviewers: They are proficient and responsible for accepting or rejecting the submitted practice questions.
Core User Need	<ol style="list-style-type: none">1. As a translation submitter, I need to see my achievements so that I can recognize my impact on translations. I need to see my submission stats and generate a badge/certificate for my contributions.2. As a translation reviewer, I need to see my achievements so that I can recognize my impact on reviewing translations. I need to see my review stats and generate a badge/certificate for my contributions.3. As a question submitter, I need to see my achievements so that I can recognize my impact on submitting practice questions. I need to see my submission stats and generate a badge/certificate for my contributions.4. As a question reviewer, I need to see my achievements so that I can recognize my impact on reviewing questions. I need to see my review stats and generate a badge/certificate for my contributions.
What goals do we want the solution to achieve?	<p>Value the impact of Translation and Question translation/question contributors(submitters and reviewers) on Oppia. The following goals are achieved by this solution.</p> <ol style="list-style-type: none">1. Recognize overall contribution for translations2. Recognize the effectiveness of translations3. Recognize overall contribution for reviewing translations4. Recognize contribution for correcting translations5. Recognize overall contribution for question submissions6. Recognize the effectiveness of submitted questions7. Recognize overall contribution for reviewing questions8. Recognize contribution for correcting question submissions

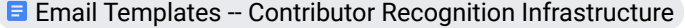
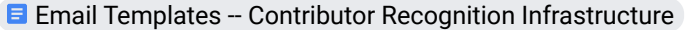
Section 2.1: WHAT

This section enumerates the requirements that the technical solution outlined in "Section 2: HOW" must satisfy.

Key User Stories and Tasks

#	Title	User Story Description (role, goal, motivation) "As a ..., I need ..., so that ..."	Priority ¹	List of tasks needed to achieve the goal (this is the "User Journey")	Links to mocks / prototypes, and/or PRD sections that spec out additional requirements.
1	Milestones celebration for translation submitters	As a translation submitter, I need to see the impact of my translations so that I can feel good about my contributions.	Must have	Get translations accepted and achieve milestones.	Translator will get a new badge.
				Goto /contributor-dashboard page and select "My Stats" and select "Badges" from the left sub-menu.	Translator will get an email after achieving a milestone.  Email Templates -- Contributor Recognition Infrastructure
2	Certificate generation for translation submitters	As a translation submitter, I need to generate a certificate so that I can show proof of my contributions.	Must have	Submit translations to see statistics/levels.	Translation submitter stats summary
				Goto /contributor-dashboard page and select "My Stats" and select "Translation Contributions". Click on the "Download" button.	Translation submitter certificate
3	Milestones celebration for question submitters	As a practice questions submitter, I need to see the impact of my question submissions so that I can feel good about	Must have	Get questions accepted and achieve milestones.	Question submitter will get a new badge.
					Question submitter will get an email after achieving a milestone.  Email Templates -- Contributor Recognition Infrastructure

¹ Use the MoSCow system ("Must have", "Should have", "Could have"). You can read more [here](#).

		my contributions.		Goto /contributor-dashboard page and select "My Stats" and select "Badges" from the left sub-menu.	Question submitter will see all of his/her badges
4	Certificate generation for question submitters	As a practice questions submitter, I need to generate a certificate so that I can show proof of my contributions.	Must have	Submit questions to see statistics/levels.	Question submitter stats summary
				Goto /contributor-dashboard page and select "My Stats" and select "Question Contributions". Click on the "Download" button.	Question submitter certificate
5	Milestones celebration for translation reviewers	As a translations reviewer, I need to see the impact of my reviews, so that I can feel good about my contributions.	Must have	Review translations and achieve milestones.	Translation reviewer will get a new badge.
					Translation reviewer will get an email after achieving a milestone. 
				Goto /contributor-dashboard page and select "My Stats" and select "Badges" from the left sub-menu.	Translation reviewer will see all of his/her badges
				Review translation to see statistics/levels.	Translation reviewer stats summary
				Goto /contributor-dashboard page and select "My Stats" and select "Translation Reviews".	
6	Milestones celebration for questions reviewers	As a practice questions reviewer, I need to see the impact of my reviews, so that I can feel good about my contributions.	Must have	Review questions and achieve milestones.	Question reviewer will get a new badge.
					Question reviewer will get an email after achieving a milestone. 
				Goto /contributor-dashboard page and select "My Stats" and select "Badges" from the left sub-menu.	Question reviewer will see all of his/her badges

				Review translation to see statistics/levels.	Question reviewer stats summary
				Goto /contributor-dashboard page and select "My Stats" and select "Question Reviews".	
7	Verification of contributors' certificates	As someone who reads the certificate of a contributor, I need to see the real statistics of the contributor so that I can verify the certificate obtained by the contributor.	Must have	Scan the QR code in the certificate.	Contributor Accomplishments Verification Page
				Redirect into the contributor accomplishments verification page.	

Technical Requirements

Additions/Changes to Web Server Endpoint Contracts

#	Endpoint URL	Request type (GET, POST, etc.)	New / Existing	Description of the request/response contract (and, if applicable, how it's different from the previous one)
1.	/contributorstatssummaries handler	GET	New	This endpoint will return data to create the summary of the user's contributions. This will basically return objects that have the following properties <ul style="list-style-type: none"> • number of submissions • number of translated words • language, topic • month of contributions • number of suggestions that are edited by the reviewer • number of accepted submissions
2.		GET		This endpoint will return a PDF file of the certificate that is requested for the given

	/contributorcertificatehandler/<contribution_type>/<contribution_sub_type>/<contributor_username>			<ul style="list-style-type: none"> • contribution_type: translations/questions • contribution_sub_type: submissions/reviews • contributor_username: the username of the contributor
3.	/contributor-dashboard-accomplishments-verification/<username>/<contribution_stats_type>	GET	New	This will render the contributor accomplishments verification page. The request will contain the contributor username and the contribution type. This will return the contributor accomplishments verification html page with contributor username and the contribution type.

Calls to Web Server Endpoints

#	Endpoint URL	Request type (GET, POST, etc.)	Description of why the new call is needed, or why the changes to an existing call is needed
1	/contributorstatssummarieshandler/<contributor_username>	GET	This endpoint call is required to fetch the user's contribution stats.
2	/contributorcertificatehandler/<contribution_type>/<contribution_sub_type>/<contributor_username>	GET	This endpoint call is required to fetch contributor certificates.
3	/contributor-dashboard-accomplishments-verification/<username>/<contribution_stats_type>	GET	This endpoint is required to render the contributor accomplishments verification page.

UI Screens/Components

#	ID	Description of new UI component	i18n required?	Mock/spec links	A11y requirements
1.	translation-submission-stats	A section that shows translation submitter's stats	-	Translation submitter stats summary	-
2.	translation-review-stats	A section that shows translation reviewer's stats	-	Translation reviewer stats summary	-

3.	question-submission-stats	A section that shows question submitter's stats	-	Question submitter stats summary	-
4.	question-review-stats	A section that shows question reviewer's stats	-	Question reviewer stats summary	-
5.	contributor-badge	A component to create badges for contributors	-	Example badge generated from the contributor-badge component	-
6.	achievement-selector	A component that works as a selector to choose the achievement type that the user wants to load.	-	Achievements selector	-
7	contributor-accomplishments-verification-page	A page that will load a particular contributor's stats when the QR code in the certificate is scanned.	-	Contributor Accomplishments Verification Page	-

Data Handling and Privacy

#	Type of data	Description	Why do we need to store this data?	Anonymized?	Can the user opt out?	Wipeout policy	Takeout policy
1.	Contributor statistics	Details about the stats of submitting/reviewing translations/questions	This allows us to display contributor's statistics and allows us to recognize their contributions.	No. In the datastore, this data is explicitly tied to a user.	No.	Delete when user deletes their account	Allow user to export this data

Other Requirements

N/A

Section 2.2: HOW

Existing Status Quo

Currently translators' accepted word counts are surfaced through the contributor admin dashboard.

The screenshot shows the Oppia Contributor Dashboard Admin interface. At the top, there is a dark blue header with the text '@ppia > Contributor Dashboard Admin' and a yellow circular icon on the right. Below the header, there are two main sections:

Remove Contribution Rights

This section contains a form with the following fields:

- Enter Username:** A text input field with the placeholder text 'Enter username'.
- Remove from:** A dropdown menu.
- Remove Rights:** A green button.

View Translation Contribution Stats

This section contains a form with the following fields:

- Enter Username:** A text input field with the placeholder text 'Enter username'.
- View Stats:** A green button.

Below the 'View Stats' button is a table with the following data:

Months	Language	Topic Name	Submitted Translation Cards	Submitted Translation Word Count	Accepted Translation Cards	Accepted Translation Cards Without Reviewer Edits	Accepted Translation Word Count
["Jul 2021"]	Portuguese	Dummy Topic 1	1	7	0	0	0

At the bottom right of the dashboard, there is a 'Success.' notification box.

But there is no mechanism for contributors to see their stats and export them to a certificate. Furthermore, there is no rewarding mechanism to recognize contributors' milestones.

Solution Overview

The solution is presented to recognize contributions of the contributors and to celebrate their milestones. Different key performance indicators will be evaluated to achieve different goals.

Goals

1. Recognize overall contribution for translations
2. Recognize the effectiveness of translations
3. Recognize overall contribution for reviewing translations
4. Recognize contribution for correcting translations
5. Recognize overall contribution for question submissions
6. Recognize the effectiveness of submitted questions
7. Recognize overall contribution for reviewing questions
8. Recognize contribution for correcting question submissions

Key Performance Indicators(KPIs)

Contributor	KPI
Translation Submitter	Number of submitted cards
	Number of accepted translations
Translation Reviewer	Number of cards reviewed
	Number of edited cards
Question Submitter	Number of submissions
	Number of accepted question submissions
Question Reviewer	Number of question reviews
	Number of questions edited by the reviewer

- Based on these KPIs, contributors will be evaluated using a **Non Linear Scale**.

Motivation

[According to this article](#), gamification is the use of game mechanics and experience design to digitally engage and motivate people to achieve their goals. Using game mechanics experience design, people are motivated to digitally engage to achieve their goals. Successful design will enhance user engagement and will be helpful to retain users as well.

Why gamification

1. To increase contributor engagement and social sharing
2. To incentivize participation without offering a tangible reward
3. To encourage users to invest more time and effort

Source:

<https://www.customerthermometer.com/customer-retention-ideas/gamification-loyalty-program/#:~:text=Gamification%20increases%20digital%20engagement%2C%20digital,into%20a%20fully%20digital%20experience.>

Gamification Techniques Used

Milestones

- Milestones record the commitment put in by contributors. Based on the number of suggestions reviewed, number of suggestions got accepted and number of suggestions corrected by the contributors, milestones are defined. Badges will be awarded to mark milestones and contributors will be notified by an email, when they achieve a new milestone. A new badge will be offered to the contributors whenever they reach a new milestone. The number of actions done for milestones is publicly displayed.

Levels

- The aim to give levels is to give a status for the contributors about how they have been performing. It will be a name rather than a numerical value. Levels will be given based on the number of submissions and reviews. The number of actions done for levels is internal.

Based on these techniques, rewarding scales are defined.

Rewarding Scale

A non-linear scale will be used for the rewarding scale.

For Levels

For Translation Submitters/Reviewers

Submitters and reviewers will be considered separately.

Condition	Level
1 <= accepted submissions or reviews < 50	Initial Contributor
50 <= accepted submissions or reviews < 200	Apprentice
200 <= accepted submissions or reviews < 600	Rising Star
600 <= accepted submissions or reviews < 2000	Expert
2000 <= accepted submissions or reviews < 5000	Master
5000 <= accepted submissions or reviews	Legend

For Question Submitters/Reviewers

Submitters and reviewers will be considered separately.

Condition	Level
1 <= accepted submissions or reviews < 15	Initial Contributor
15 <= accepted submissions or reviews < 45	Apprentice
45 <= accepted submissions or reviews < 150	Rising Star
150 <= accepted submissions or reviews < 600	Expert
600 <= accepted submissions or reviews < 1500	Master

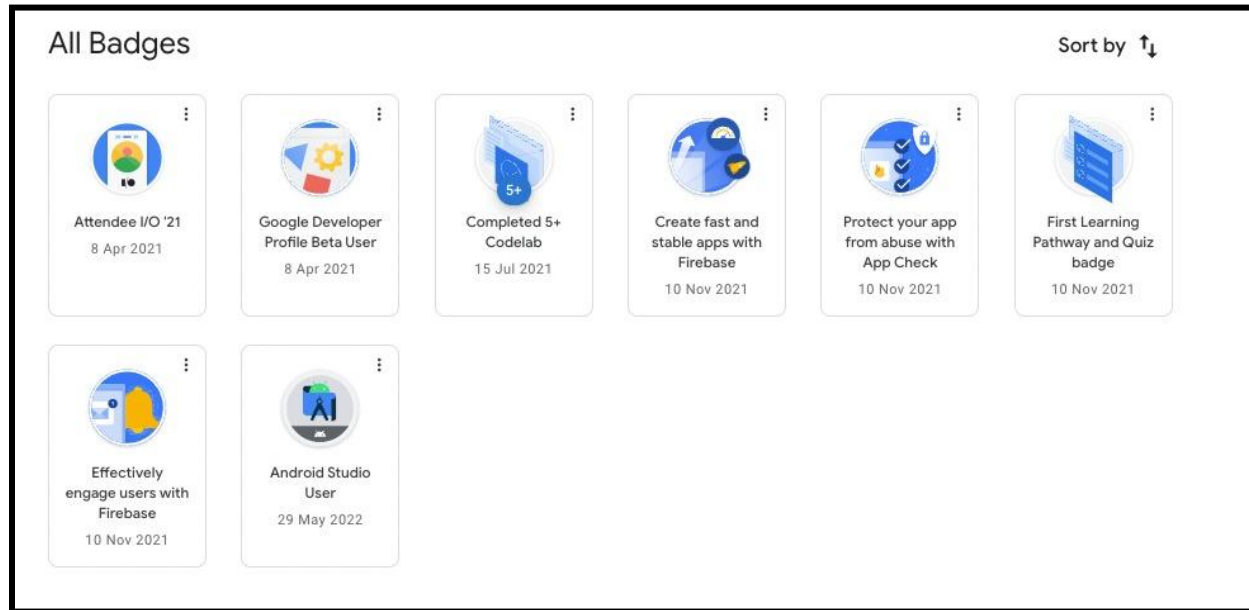
1500 <= accepted submissions or reviews

Legend

For Milestones

Different approaches by different products for milestones celebrations

- Google developer profile shows the badges achieved by developers as below



- But it does not show the upcoming milestones/badges since for a certain category only one badge is offered.
- They have the option of favouriting badges — we can not use this feature since we do not explicitly store data related to badges.
- [At huddo](#), it shows different levels of badges for a given category as below. Users can have an idea of what they need to do in order to achieve the next badge/milestone.

Profiles

MY PROFILE MY NETWORK DIRECTORY

Adam Brown [Network Contact](#)
 Director of Innovation & Strategy

Send Email Remove From Network Stop Following Share a File Download vCard

Recent Updates Contact Information Background Huddo Badges Huddo Thanks **Huddo Awards** Communities I Own

Profile Progress
 100% - Congratulations on completing your profile!

Tags
 Add tag(s) to this profile
 connections development domino gamification ISW jp kudos lotus marketing nsw ohs portal qis quickr sales sametime social social-media tas tdi vic websphere wildfire
 Cloud | List

Things in Common
 Activity (1)
 CCM Migrator Develop...
 Files (2)

Achiever
 1 of 6

Culture
 0 of 4

Loyalty
 5 of 7

Superstar
 0 of 5

Achiever

Culture

Loyalty

- Badges/achievements of every category are shown in one place
- At [runsignup](#), they show last achieved badges in the profile page, and a complete description including previously achieved badges and future milestones are shown in a separate screen/page called “trophy case”.



Sarah Kim
The Warm Up Fitness
Challenge - Wave 1

[Photos](#) [Certificate](#)

[Log Activities](#)

107

Team Results: You Are On Mute

Tally
2450

Overall Progress
350%

Overall
19 of 204

[Questions About Results?](#)

[Trophy Case](#)

Overall

2450 of 700

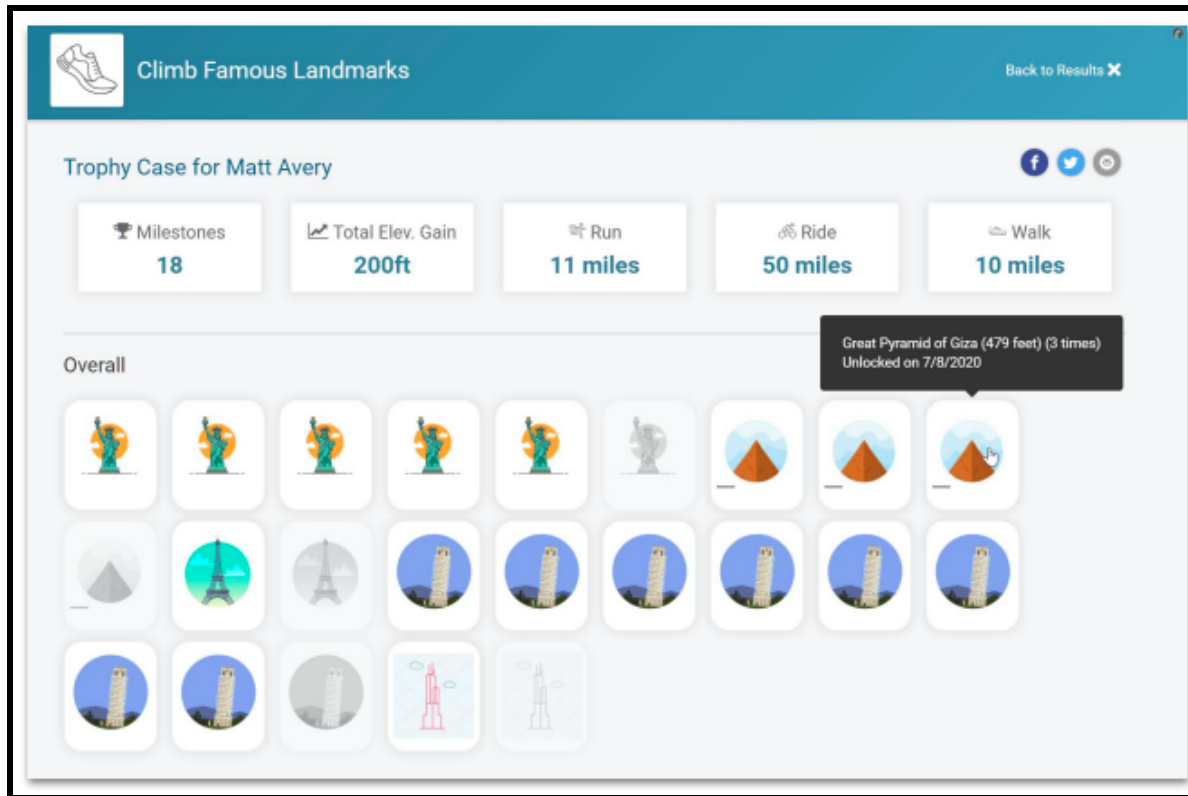
Progress 350%



Photos

[View All Photos](#)



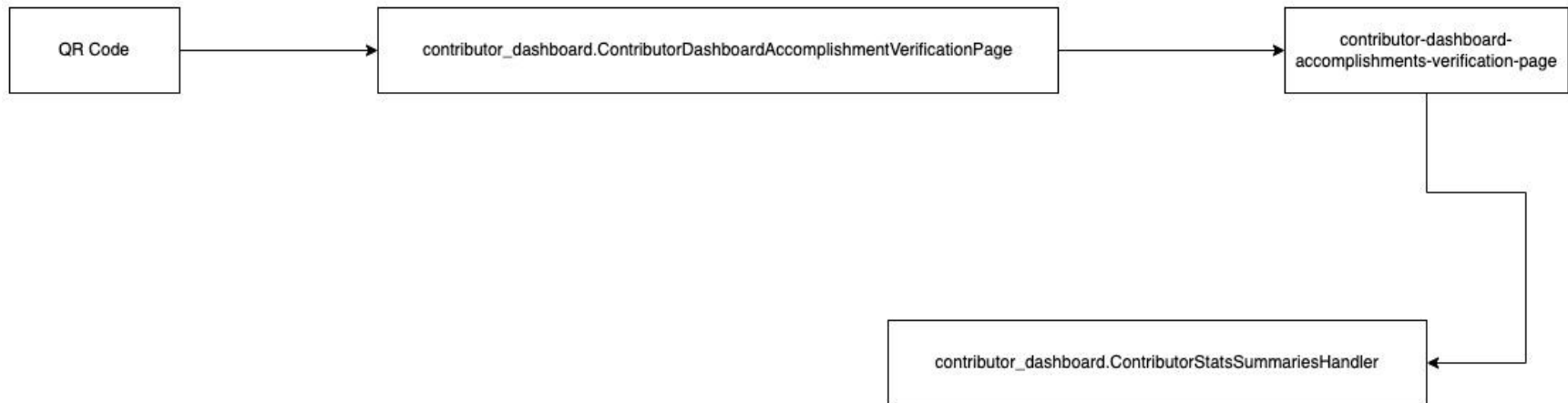


	Google Badges	Huddo	runsignup
Ability to pin badges	Available	Not available	Not available
Showing upcoming badges	Not available	Available	Available
Different levels of badges for the same category	Not available	Available	Available

For Oppia contributor dashboard, I have considered an approach which is based on as a combination of what huddo and runsignup does.

- Contributors will earn new badges, if they submit/review 'x' number of suggestions.
- Contributors will receive new milestones for:
 - reviewing: 1/10/20/50/100/500/1000
 - getting submissions accepted: 1/10/20/50/100/500/1000
 - for submitting: 1/10/20/50/100/500/1000
 - correcting submissions: 1/10/20/50/100/500/1000

Verification of Contributor Certificate



- TranslationContributionStats are stored in a separate model. This will include stats regarding translation contributions. There will be one entry per (language code, contributor user ID, topic ID) tuple. It will have the following fields.
 - language_code: str. indexed.
 - contributor_user_id: str. indexed.
 - topic_id: str. indexed.
 - submitted_translations_count: int. indexed.
 - submitted_translation_word_count: int. indexed.
 - accepted_translations_count: int. indexed.
 - accepted_translations_without_reviewer_edits_count: int. indexed.
 - accepted_translation_word_count: int. indexed.
 - accepted_translations_without_reviewer_edits_word_count: int. indexed.
 - contribution_months: list(str). indexed.
 - first_contribution_date: date. indexed.
 - last_contribution_date: date. indexed.
- TranslationReviewStats are stored in a separate model. This will include stats regarding question contributions. There will be one entry per (language code, contributor user ID, topic ID) tuple. It will have the following fields.
 - language_code: str. indexed.
 - contributor_user_id: str. indexed.
 - topic_id: str. indexed.
 - reviewed_translations_count: int. indexed.
 - reviewed_translation_word_count: int. indexed.
 - accepted_translations_count: int. indexed.
 - accepted_translations_with_reviewer_edits_count: int. indexed.
 - accepted_translations_with_reviewer_edits_word_count: int. indexed.
 - review_months: list(str). indexed.
 - first_contribution_date: date. indexed.
 - last_contribution_date: date. indexed.
- QuestionContributionStats are stored in a separate model. This will include stats regarding question contributions. There will be one entry per (contributor user ID, topic ID) tuple. It will have the following fields.
 - contributor_user_id: str. indexed.
 - topic_id: str. indexed.
 - submitted_questions_count: int. indexed.

- accepted_questions_count: int. indexed.
- accepted_questions_without_reviewer_edits_count: int. indexed.
- contribution_months: list(str). indexed.
- first_contribution_date: date. indexed.
- last_contribution_date: date. indexed.
- QuestionReviewStats are stored in a separate model. This will include stats regarding question reviews. There will be one entry per (contributor user ID, topic ID) tuple. It will have the following fields.
 - contributor_user_id: str. indexed.
 - topic_id: str. indexed.
 - reviewed_questions_count: int. indexed.
 - accepted_questions_count: int. indexed.
 - accepted_questions_with_reviewer_edits_count: int. indexed.
 - review_months: list(str). indexed.
 - first_contribution_date: date. indexed.
 - last_contribution_date: date. indexed.

Third-Party Libraries

No.	Third-party library name and version	Link to third-party library	Why it is needed	License ² (if third-party library)
1	pdftkit	https://pypi.org/project/pdftkit/	It is required to generate contributor certificates(pdf).	MIT

“Service” Dependencies

N/A

²Note: Oppia can only use third-party libraries that are compatible with our Apache 2.0 license. If you're unsure about license compatibility, talk to a platform TL.

Impact on Other Oppia Teams

N/A

Key High-Level and Architectural Decisions

Decision 1: How contributors are valued

I have considered the following alternatives.

1. Milestones: Whenever contributors achieve a predefined goal, they will reach a milestone.
2. Streaks: An uninterrupted sequence of submitting suggestions.

Among these, I believe that Alternative 1 is the best approach, because:

- Contributors are not forced to make their contributions and they can take decisions on how/when they need to contribute.
- Their level will steadily increase rather than start all over again like in streaks where they will have to start all over again once a streak is over.

	Alternative 1	Alternative 2
Contributor retention	Possible	Possible
Ability to take breaks when they needed/due to unavoidable circumstances	Possible — It won't affect their achieved milestones.	Not possible — Streak will be broken if the break is for too long.
Steady progression of achievements	Possible	Not possible — since once a streak gets over, contributors will have to start all over again from the beginning. This would frustrate the contributor thinking that why should I start all over again right from the beginning and even would quit contributing.
Forcing the contributor to keep	Not forced	Will have to force — If the contributor is

contributing		suddenly inactive within a streak, they should be notified that they need to come back and keep contributing so that they would not lose their streak.
--------------	--	--------------------------------------------------------------------------------------------------------------------------------------------------------

Decision 2: How to measure milestones

I have considered the following alternatives:

1. Measure based on their raw stats: milestones are offered like the following.
 - a. For submitting 1/10/20/50/100/500/1000 submissions
 - b. For getting accepted 1/10/20/50/100/500/1000 submissions
 - c. For reviewing 1/10/20/50/100/500/1000 submissions
 - d. For correcting 1/10/20/50/100/500/1000 submissions
2. Measure from a scoring algorithm: scoring algorithm is based on their stats and the value taken from an equation should be considered.
 - a. For reaching a score of 5
 - b. For reaching a score of 10
 - c. For reaching a score of 20
 - d. For reaching a score of 50
 - e. For reaching a score of 100

Among these, I believe that the Alternative 1 is the best approach, because:

- contributors can easily understand how they can achieve new milestones.

	Alternative 1	Alternative 2
Complexity to understand	Low — since contributors know how exactly they can achieve a milestone	High — contributors do not know the underlying algorithm so that they do not know how much of exact suggestions that they need to submit/review to achieve milestones.
Unified way to recognize all stats	Not possible — each statistic such as no	Possible — since everything is included in

	of submissions, no of accepted submissions without reviewer edits should be treated separately.	the algorithm everything can be considered from the equation
--	-------------------------------------------------------------------------------------------------	--------------------------------------------------------------

Decision 3: Level/Milestone scale type

I have considered the following alternatives:

1. Linear scale: the difference between each level/milestone is a constant
 - Ex: a new level/milestone is achieved in each 5 submission/review.
2. Non-linear scale: the difference between each level/milestone is increased level by level.

Among these, I believe that the Alternative 2 is the best approach, because:

- achieving new levels/milestones becomes comparatively difficult.
- the pace that we would receive submission/reviews would increased.

	Alternative 1	Alternative 2
Difficulty to achieve new levels/milestones	Comparatively easy – once contributors become proficient in what they do, they would submit/review suggestions quickly so that they would easily achieve new levels/milestones. This would also make the contributor bored since there would not be an enthusiasm to achieve a new levels/milestone.	Comparatively difficult – since the gap increases between each levels/milestone the pace that they would achieve would be reduced.
Pace of receiving new submissions/reviews	Constant – since the gaps between levels/milestones are equal, no need to be hurry to reach new levels/milestones.	Comparatively fast – since contributors need to submit/review more suggestions to reach new levels/milestones, they need to submit/review suggestions quickly if they have the same interest.

Decision 4: How stats are initialized/updated

I have considered the following alternatives:

1. Initialize/update them on the fly
2. Initialize/update them via a daily cron job

Among these, I believe that the Alternative 1 is the best approach, because:

- When using a cron job, contributors can not see their stats in real time.

The above approaches are contrasted in detail in the following table:

	Alternative 1	Alternative 2
Ability to see stats in real time	Possible	Not possible – since we do it via a cron job(once per day) contributor's statistics are not available to see in real time. This may confuse them.
Complexity	High	Low – implementation is straightforward to calculate stats from storage models.

Decision 5: Where badges are displayed

I have considered the following alternatives:

1. Separate stats and badges in separate tabs — [Overview of Stats, Badges](#)
2. Display them with the stats — in the left side pane before the stats table

Among these, I believe that the Alternative 1 is the best approach, because:

- Badges can be arranged neatly rather than keeping them in a smaller space in the stats page.

	Alternative 1	Alternative 2
Arrangement of the badges	Can be arranged neatly – One contributor can approximately achieve 42 badges if he/she has all the roles.	Can not be arranged neatly because the space is limited when the size of the badge and the number of badges are considered.

Direct visibility of badges together with stats	Not possible	Possible
-------------------------------------------------	--------------	----------

Decision 6: How badges of different categories(Translation submitter/reviewer, question submitter/reviewer) are displayed

I have considered the following alternatives:

1. Display all at once
2. Display badges by the selected category according to the selected contribution type from the “Contribution Type” dropdown.

Among these, I believe that the Alternative 1 is the best approach, because:

- contributors can see all of their achieved/upcoming badges in one place so that they can have an idea of overall milestones that they achieved.

	Alternative 1	Alternative 2
Users can have an idea of overall milestones that they achieved.	Possible	Not possible
Space taken to show badges	Comparatively large	Comparatively small

Decision 7: How upcoming milestones are displayed

I have considered the following alternatives:

1. Show only the next badge(disabled) that they can obtain.
2. Show all the badges that they can achieve and disable the badges that they have not unlocked.

Among these, I believe that the Alternative 1 is the best approach, because:

- showing all the badges that they can achieve may demotivate them to progress.

	Alternative 1	Alternative 2
Users may demotivate to progress by seeing the overhead.	Not possible	Possible — since it shows all the milestones that they need to achieve(like

		1000 submissions) in order to unlock, it may demotivate the user.
Users can get an idea what they need to do achieve in order to achieve next milestone(s)	Just the next milestone can be shown	Possible

Decision 8: How badges for translations are differentiated for different languages

I have considered the following alternatives:

1. Only use the language code in the badge.
2. Include an image/symbol in the badge.

Among these, I believe that the Alternative 1 is the best approach, because:

- it doesn't require more storage.
- it can differentiate badges based on languages.

	Alternative 1	Alternative 2
Ability to give a uniqueness based on the language for the badge	Possible – The language name is included in the badge so that the users can know for which language the badge is offered.	Possible – Every language has a unique image so that users can clearly differentiate the badges obtained for different languages.
Extra storage	Not required – since the language name can be programmatically set, no need for extra storage.	Required – there are no unique icons that can differentiate languages so that we should store new images related to different languages.

Decision 9: Choosing a KPI for levels

I have considered the following alternatives:

1. Number of submissions/reviews
2. Number of words submitted/reviewed

Among these, I believe that the Alternative 1 is the best approach, because:

- Contributors have to make more submissions/reviews to achieve higher levels. This will eliminate them from always making short submissions/reviews.
- The length of the submission/review is not always a good KPI for both translations and questions.

	Alternative 1	Alternative 2
Contributor's tendency to make short submissions/reviews	Possible – But still in order to achieve higher levels, contributors have to submit/review a considerable amount of submissions/reviews. When they go for more submissions/reviews length of submissions/reviews should be properly distributed.	Not possible
Generalize for both questions and translations	Possible	Not possible – Having more words within a question does not always mean it is a strong question. There can be good, strong questions that may have text input interactions. There can also be short questions that would strongly test skill. So the number of words can not be used as a good KPI for both translations and questions.

Decision 10: Shape of the badge


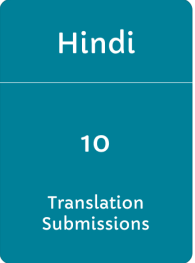


I have considered the following alternatives.

1. Circular shape
2. Rectangular shape
3. Hexagonal shape
4. A combination of multiple shapes — designed based on [Free Awards & Achievements Icons Set by Alex Martynov](#)

Among these, I believe that the Alternative 4 is the best approach, because:

- Ability of programmatically set texts

- the impressive look of the design



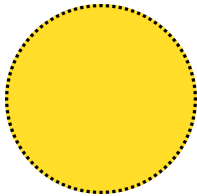
	Alternative 1	Alternative 2	Alternative 3	Alternative 4
				
Ability of programmatically set texts	Difficult to align text to curvature.	Easy	Easy	Easy
Visibility of texts	Less visible because texts are aligned with the curvature along with stars.	Properly visible	Properly visible	Properly visible

Decision 11: Designs for upcoming badges

I have considered the following alternatives.

1. Reduced opacity with a lock to the badge — references: <https://helpx.adobe.com/mt/captivate-prime/learners/feature-summary/badges.html>
2. Dotted outlines to the badge

	Alternative 1	Alternative 2	Alternative 3

			
Feeling that this is not an already achieved badge	High — reduced opacity gives a feeling like this is not a real badge.	Low — since the color combination between this and the original badge is almost the same. Also dots do not seem to be matching perfectly with the ribbon.	High — since there is no decorations as of the original badge.
Feeling that this should be unlocked	High — reduced opacity and the lock suggest it.	Low — only the question mark suggest that this needs to be unlocked.	Low — no suggestion that this needs to be unlocked.

Decision 12: Email notification trigger method

I have considered the following alternatives.

1. Sending emails through daily cron job
2. Sending emails on the fly whenever a contributor performs a specific action and achieves a new milestone — using a queuing strategy to send real time emails.

Among these, I believe that the Alternative 2 is the best approach, because:

- We can reduce the complexity and get rid of the new storage model.

	Alternative 1	Alternative 2
Effect for usual contributor dashboard workflows such translation submissions/reviews and question submissions/reviews	None	There is an effect. But this can be reduced if we send these emails asynchronously.

Having to maintain a separate model to track last achieved milestones	Required	Not required.
-----------------------------------------------------------------------	----------	---------------

Decision 13: Contributor certificate generation

I have considered the following alternatives.

1. Generate them in the backend
2. Generate them in the frontend

Among these, I believe that the Alternative 1 is the best approach, because:

- More protection over faking of the certificate.

	Alternative 1	Alternative 2
Complexity	Moderate — will have to add new endpoints to create data related to certificates.	Medium — can be generated from the existing stats received when loading the stats page.
Ability to change real data and fake the certificate	Low — since everything is generated from the backend	Considerable — Certificate can not be faked easily since the library will not create the PDF from an HTML file but with the inputs it gets, so that the users can not just fake the certificate by changing HTML tags. There will be a certificate verification implementation in the future so that there won't be any faking of the certificates.

Decision 13: Contributor certificate design

I have considered the following alternatives.

1. [Alternative 1](#)
2. [Alternative 2](#)

Among these, I believe that the Alternative 1 is the best approach because:

- It looks more of a certificate rather than a letter.

	Alternative 1	Alternative 2
The more look/feel as a certificate	It looks more of a certificate	It looks more of a letter.

Risks and mitigations

N/A

Implementation Approach

[Web only] Storage Model Layer Changes

New storage stats models will be added to the `core/storage/suggestion/gae_models.py`:

Note: Existing TranslationContributionStatsModel will be reused.

TranslationReviewStatsModel

- **Constraints:**
 - One entry per (language code, contributor user ID, topic ID) tuple.
 - All fields required
- **ID generation**
 - `[language_code].[contributor_user_id].[topic_id]`
 - According to this; user id can not contain a period
https://github.com/oppia/oppia/blob/6fe952c4d88b6442a1643f6a4faf4f8e56becbb3/core/storage/user/gae_models.py#L311
 - Topic id is created by this `get_id` function
https://github.com/oppia/oppia/blob/86df5d4e5378365cf229684e8ea89ef9e766ac74/core/storage/base_model/gae_models.py#L468

- This will call the `convert_to_hash` function which will return an alphanumeric value. <https://github.com/oppia/oppia/blob/86df5d4e5378365cf229684e8ea89ef9e766ac74/core/utils.py#L503>
- Hence topic ID also should not contain periods
- Language codes will obviously not have periods
- **Fields:**
 - `language_code`: str. indexed.
 - `contributor_user_id`: str. indexed.
 - `topic_id`: str. indexed.
 - `reviewed_translations_count`: int. indexed.
 - `reviewed_translations_word_count`: int. indexed.
 - `accepted_translations_count`: int. indexed.
 - `accepted_translations_with_reviewer_edits_count`: int. indexed.
 - `accepted_translations_with_reviewer_edits_word_count`: int. indexed.
 - `review_months`: list(str). indexed.
 - `first_contribution_date`: date. indexed.
 - `last_contribution_date`: date. indexed.
- **Validations:**
 - Language code is supported
 - Accepted translation counts \leq Reviewed translation counts
 - All counts are non-negative
- **Deletion Policy:** DELETE
- **Takeout Policy:**
 - `get_model_association_to_user`: ``base_models.MODEL_ASSOCIATION_TO_USER.MULTIPLE_INSTANCES_PER_USER``
 - `get_export_policy`: All fields except ``contributor_user_id`` are exported.

QuestionContributionStatsModel

- **Constraints:**
 - One entry per (contributor user ID, topic ID) tuple.
 - All fields required.
- **ID generation**
 - `[contributor_user_id].[topic_id]`
- **Fields:**

- contributor_user_id: str. indexed.
- topic_id: str. indexed.
- submitted_questions_count: int. indexed.
- accepted_questions_count: int. indexed.
- accepted_questions_without_reviewer_edits_count: int. indexed.
- contribution_months: list(str). indexed.
- first_contribution_date: date. indexed.
- last_contribution_date: date. indexed.
- **Validations:**
 - Accepted question counts < Submitted question counts
 - All counts are non-negative
- **Deletion Policy:** DELETE
- **Takeout Policy:**
 - *get_model_association_to_user*: `base_models.MODEL_ASSOCIATION_TO_USER.ONE_INSTANCE_PER_USER`
 - *get_export_policy*: All fields except `contributor_user_id` are exported.

QuestionReviewStatsModel

- **Constraints:**
 - One entry per (contributor user ID, topic ID) tuple.
 - All fields required.
- **ID generation**
 - [contributor_user_id].[topic_id]
- **Fields:**
 - contributor_user_id: str. indexed.
 - topic_id: str. indexed.
 - reviewed_questions_count: int. indexed.
 - accepted_questions_count: int. indexed.
 - accepted_questions_with_reviewer_edits_count: int. indexed.
 - review_months: list(str). indexed.
 - first_contribution_date: date. indexed.
 - last_contribution_date: date. indexed.

- **Validations:**
 - Accepted question counts < Submitted question counts
 - All counts are non-negative
- **Deletion Policy:** DELETE
- **Takeout Policy:**
 - *get_model_association_to_user*: `base_models.MODEL_ASSOCIATION_TO_USER.ONE_INSTANCE_PER_USER`
 - *get_export_policy*: All fields except `contributor_user_id` are exported.

Domain Objects

core/domain/suggestion_registry.py

TranslationReviewStats

- language_code: str.
- topic_id: str.
- contributor_user_id: str.
- reviewed_translations_count: int.
- reviewed_translation_word_count: int.
- accepted_translations_count: int.
- accepted_translations_with_reviewer_edits_count: int.
- accepted_translations_with_reviewer_edits_word_count: int.
- review_months: list(str). indexed.
- first_contribution_date: date.
- last_contribution_date: date.

QuestionContributionStats

- contributor_user_id: str.
- topic_id: str.
- submitted_questions_count: int.
- accepted_questions_count: int.
- accepted_questions_without_reviewer_edits_count: int.
- contribution_months: list(str). indexed.

- first_contribution_date: date.
- last_contribution_date: date.

QuestionReviewStats

- contributor_user_id: str.
- topic_id: str.
- reviewed_questions_count: int.
- accepted_questions_count: int.
- accepted_questions_with_reviewer_edits_count: int.
- review_months: list(str). indexed.
- first_contribution_date: date.
- last_contribution_date: date.

ContributorMilestoneEmailInfo

- contributor_user_id: str.
- contribution_type: str.
- contribution_sub_type: str.
- language_code: str|None.
- contributions_count: int.

ContributorStatsSummary

- contributor_user_id: str.
- translation_contribution_stats: List[TranslationReviewStats]
- question_contribution_stats: List[QuestionContributionStats]
- translation_review_stats: List[TranslationReviewStats]
- question_review_stats: List[QuestionReviewStats]

User Flows (Controllers and Services)

Services

core/domain/suggestion_services.py

`_get_all_translation_review_stats(contributor_username)`

- *Arguments:*
 - contributor_username: str. Username of a translation reviewer.
- *Returns:*
 - list(TranslationReviewStats). List of all TranslationReviewStats for the supplied contributor username that have non-zero translation review counts.

`_get_all_question_contribution_stats(contributor_username)`

- *Arguments:*
 - contributor_username: str. Username of a question contributor.
- *Returns:*
 - list(QuestionContributionStats). List of all QuestionContributionStats for the supplied contributor username that have non-zero question submission counts.

`_get_all_question_review_stats(contributor_username)`

- *Arguments:*
 - contributor_username: str. Username of a question reviewer.
- *Returns:*
 - list(QuestionReviewStats). List of all QuestionReviewStats for the supplied contributor username that have non-zero question review counts.

`get_all_contributor_stats_summaries(contributor_username)`

- *Arguments:*
 - contributor_username: str. Username of a contributor.
- *Returns:*
 - ContributorStatsSummary. The corresponding ContributorStatsSummary object.

`_update_contributor_stats(suggestion, contributor_username, topic_id)`

- *Arguments:*
 - suggestion: Suggestion. Suggestion that will be in review/accepted/rejected.
 - contributor_username: str. Username of a contributor.
 - topic_id: str. The ID of the topic that the suggestion is made for.

`_enqueue_contributor_achievement_notification_email_task(contributor_milestone_email_info)`

- *Arguments:*

- contributor_milestone_email_info: ContributorMilestoneEmailInfo. An object that contains data to send the contributor's milestone notification email.
- This will call the `taskqueue_services.enqueue_task()` to send the email asynchronously.
 - Example:
 - https://github.com/oppia/oppia/blob/f35a6638b0d3f483c87106fcb8e7ba2ac21f6238/core/domain/feedback_services.py#L864

`_calculate_submitter_milestone_achievements(contributor_username, contribution_type)`

- *Arguments:*
 - contributor_username: str. Username of a contributor.
 - contribution_types: str. Type of the contribution i.e. translation/question.
- This method will calculate whether a submitter has achieved a new milestone and will call `_enqueue_contributor_achievement_notification_email_task` to send an email to notify the submitter.

`_calculate_reviewer_milestone_achievements(contributor_username, contribution_type, edited_by_reviewer)`

- *Arguments:*
 - contributor_username: str. Username of a contributor.
 - contribution_types: str. Type of the contribution i.e. translation/question.
 - edited_by_reviewer: int. Whether the suggestion is edited by the reviewer.
- This method will calculate whether a reviewer has achieved a new milestone and will call `_enqueue_contributor_achievement_notification_email_task` to send an email to notify the reviewer.

`generate_contributor_certificate(contributor_username, contribution_type, contribution_sub_type)`

- *Arguments:*
 - contribution_type: str.
 - This can be translation or question
 - contribution_sub_type: str.
 - This can be submissions/reviews
 - contributor_username: str.
 - The username of the corresponding contributor.
- This method will generate the contributor certificate based on the stats. **pdfkit** library will be used to generate PDF files.
 - Fetch stats from stats models based on the `contribution_type` and `contribution_sub_type`
 - if `contribution_type = translation` and `contribution_sub_type = submissions`:

- fetch from the TranslationContributionStatsModel
 - if contribution_type = question and contribution_sub_type = submissions:
 - fetch from the QuestionContributionStatsModel
- Create data required for the certificates – Will use `pdftk.from_string` to write to PDF file.
 - [Translation submitter's certificate](#)



CERTIFICATE OF APPRECIATION

WE GRATEFULLY ACKNOWLEDGE

hasitha

for your dedication and time in translating Oppia's basic maths lessons to Hindi which will help our Hindi-speaking learners better understand the lessons.

This certificate confirms that hasitha has contributed x hours' worth* of translations from mon-yyyy to mon-yyyy

SIGNATURE

DATE

* This is an approximate value
Certificate verification can be done by emailing anubhuti@oppia.org, if needed.

- initial contribution date

- last contribution date
 - language
 - contribution hours
- [Question submitter's certificate](#)



- last contribution date
- contribution hours

core/domain/email_manager.py

send_contributor_milestone_email(contributor_milestone_email_infos)

- *Arguments:*
 - contributor_milestone_email_infos: list(ContributorMilestoneEmailInfo). A list of contributor milestone email information that consists of the contributor and his/her milestone.

Controllers

ContributorStatsSummariesHandler

- A new endpoint to fetch contributor's stats summaries.
- URL: /contributorstatssummarieshandler/<contributor_username>
- GET:
 - *Decorator:* @ can_access_contributor_stats
 - This will check whether the user is logged in.
 - Logged in user has the provided contributor_username in the URL
 - *Params:*
 - contributor_username: str.
 - *Data Response:* a dictionary representation of ContributorStatsSummary
 - Returns a dictionary representation of ContributorStatsSummary according to the contributor_username provided.
 - *Functionality:*
 - Calls `suggestion_services.get_all_contributor_stats_summaries()`
 - *Datastore calls:*
 - Fetch: UserModel
 - Retrieve user ID for username
 - Fetch: TopicModel
 - Retrieve topic name for topic ID

- Fetch:
TranslationContributionStatsModel/TranslationReviewStatsModel/QuestionContributionStatsModel/QuestionReviewStatsModel for the get_all_contributor_stats_summaries provided.

ContributorCertificateHandler

- A new endpoint to generate contributor certificates.
- URL: /contributorcertificatehandler/<contribution_type>/<contribution_sub_type>/<contributor_username>
- GET:
 - *Decorator:* @can_access_contributor_stats
 - This will check whether the user is logged in.
 - Logged in user has the provided contributor_username in the URL
 - *Params:*
 - contribution_type: str.
 - This can be translation or question
 - contribution_sub_type: str.
 - This can be submissions/reviews
 - contributor_username: str.
 - The username of the corresponding contributor.
 - *Data Response:* a PDF file of the certificate.
 - Returns a PDF file for the corresponding contributor for given contribution type and the contribution sub type.
 - *Functionality:*
 - Calls `suggestion_services.generate_contributor_certificate()`
 - *Datastore calls:*
 - Fetch: UserModel
 - Retrieve user ID for username
 - Fetch:
TranslationContributionStatsModel/TranslationReviewStatsModel/QuestionContributionStatsModel/QuestionReviewStatsModel according to the contribution_type and contribution_sub_type for the user.

SuggestionHandler

- An existing endpoint that handles operations relating to suggestions(submissions).
- URL: */suggestionhandler*
- The request or response is unchanged for this endpoint. It calls `suggestion_services.create_suggestion()` to save the suggestion. When this endpoint is called it will additionally call the **`_update_contributor_stats()`** from the `suggestion_services.py` to create/update contribution stats.

SuggestionToExplorationActionHandler

- An existing endpoint that actions performed on suggestions to explorations.
- URL: */suggestionactionhandler/exploration/<target_id>/<suggestion_id>*
- The request or response is unchanged for this endpoint. It calls `suggestion_services.accept_suggestion()` or `suggestion_services.reject_suggestion()` to accept/reject the suggestion. When this endpoint is called it will additionally call the **`_update_contributor_stats()`** from the `suggestion_services.py` to create/update contribution stats.
 - Compute the contribution type from the suggestion (suggestion type + suggestion status).
 - Compute which counts to update based on the suggestion status (in-review/accepted/rejected).
 - Increment by 1 the corresponding counts.
 - Add month if needed in the format of ["2022-03", "2022-05"]
 - Check whether the month that the action the user does(submission/review) is included in the `contribution_months` or `review_months`; if it doesn't exist, add the current month in the form of "yyyy-mm" to the list.

SuggestionToSkillActionHandler

- An existing endpoint that actions performed on suggestions to skills.
- URL: */suggestionactionhandler/skill/<target_id>/<suggestion_id>*
- The request or response is unchanged for this endpoint. It calls `suggestion_services.accept_suggestion()` or `suggestion_services.reject_suggestion()` to accept/reject the suggestion. When this endpoint is called it will additionally call the **`_update_contributor_stats()`** from the `suggestion_services.py` to create/update contribution stats.

[Web only] Web frontend changes

- For existing service
core/templates/pages/contributor-dashboard-page/services/contributor-dashboard-backend-api.service.ts
 - Add new endpoints to call the ContributorStatsSummariesHandler(/contributorstatssummarieshandler/<contributor_username>).
 - getContributorStatsAsync() to call /contributorstatssummarieshandler/<contributor_username>
 - Add a new endpoint to call the ContributorCertificateHandler(/contributorcertificatehandler/<contribution_type>/<contributor_username>)
 - getContributorCertificateAsync() to call /contributorcertificatehandler/<contribution_type>/<contributor_username>
- Create a new component core/templates/pages/contributor-dashboard-page/contributor-stats to display the new tab “My Stats”. This component will display different stats and the badges earned by the contributor. Data for the badges will be computed in real time using the stats.
 - Data required to compute badges
 - For translation badges
 - total acceptances in each language – for submitters
 - total reviews in each language – for reviewers
 - total corrections in each language – for reviewers
 - For question badges
 - total acceptances – for submitters
 - total reviews – for reviewers
 - total corrections – for reviewers
 - Iterate over the milestone descriptions defined in [the constants file](#).
 - Check each milestone achieved according to the data calculated.
 - Check what is the next milestone to be achieved and how many actions needed to achieve the next milestone. These actions mean how many acceptances/reviews/corrections are needed to achieve the next milestone so that we can show it in the tooltip when a user hovers over the disabled button.
 - Create data to build badges
 - contributionType: String. – the main type of the contribution.
 - translation

- question
- contributionSubType: String. – the kind of contributions made a given contribution type.
 - acceptances
 - reviews
 - corrections
- languageCode: String|None. – the language code for translation type contribution.
- numberOfContributions: Integer|None.
 - the number of actions(acceptances/reviews/corrections) performed to obtain a badge for unlocked badges.
 - this will be None when it is an upcoming badge i.e. badgeLocked = True
- numberOfContributionsRemaining: Integer|None.
 - the number of actions(acceptances/reviews/corrections) needs to be performed to obtain the locked badge i.e. when badgeLocked = True.
 - this will be None when it is an already achieved badge i.e. badgeLocked = False.
- badgeLocked: boolean. – whether the badge is already achieved or not.



Overview

Badges

Hello, hasitha

Explore the impact of your contributions!

Contribution Type Translation Contributions

Translation Contributions - Hindi

Initial Contributor

Months	Topic Names	Accepted Cards	Accepted Word Count
Jul 2021 - Jun 2021	Dummy Topic 1	7	150
Jul 2021 - Jun 2021	Dummy Topic 2	1	10

Download certificate

Translation Contributions - Arabic

Rising Star

Months	Topic Names	Accepted Cards	Accepted Word Count
Jul 2021 - Jun 2021	Dummy Topic 11	250	2673
Jul 2021 - Jun 2021	Dummy Topic 22	30	478

Download certificate

[Translation Contribution Stats](#)



- Overview
- Badges

Hello, hasitha

Explore the impact of your contributions!

Contribution Type Translation Reviews

Translation Reviews - Hindi

Initial Contributor

Months	Topic Names	Reviewed Cards	Reviewed Word Count	Accepted Cards	Accepted Word Count
Jul 2021 - Jun 2021	Dummy Topic 1	10	170	7	150
Jul 2021 - Jun 2021	Dummy Topic 2	2	20	1	10

[Download certificate](#)

Translation Reviews - Arabic

Initial Contributor

Months	Topic Names	Reviewed Cards	Reviewed Word Count	Accepted Cards	Accepted Word Count
Jul 2021 - Jun 2021	Dummy Topic 1	10	170	7	150
Jul 2021 - Jun 2021	Dummy Topic 2	2	20	1	10

[Download certificate](#)

[Translation review stats](#)

[My Contributions](#)[Submit Question](#)[Translate Text](#)[My Stats](#)[Overview](#)[Badges](#)

Hello, hasitha

Explore the impact of your contributions!

Contribution Type

Question Contributions

Question Contributions

Initial Contributor

Month	Topic Names	Accepted Questions	Accepted Questions Without Edits
Jul 2021 - Jun 2021	Dummy Topic 1	0	0
Jul 2021 - Jun 2021	Dummy Topic 1	0	0

[Download certificate](#)

[Question contribution stats](#)



My Contributions

Submit Question

Translate Text

My Stats

Overview

Badges

Hello, hasitha

Explore the impact of your contributions!

Contribution Type

Question Reviews ▾

Question Reviews

Initial Contributor 

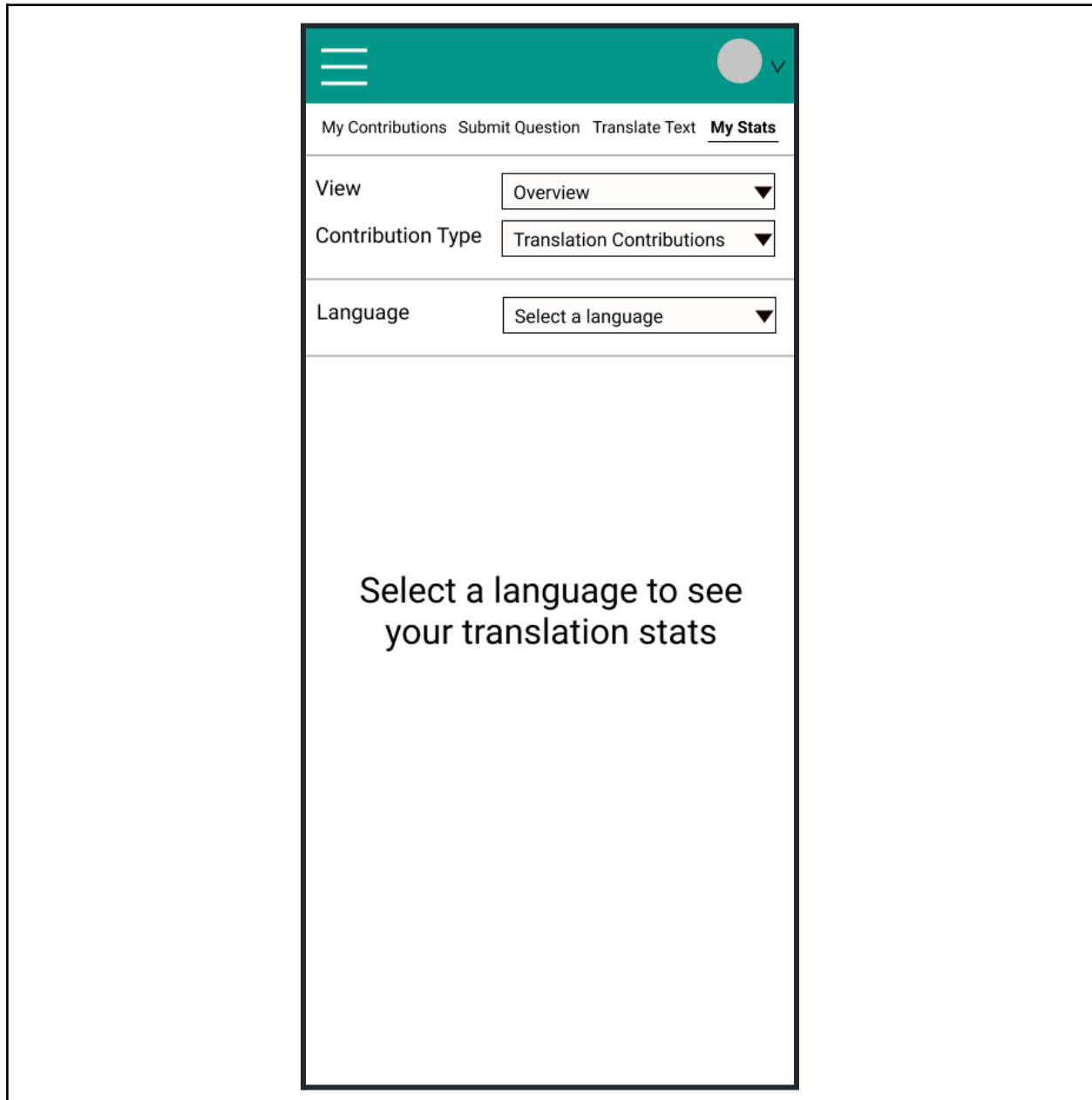
Month	Topic Names	Reviewed Questions	Accepted Questions
Jul 2021 - Jun 2021	Dummy Topic 1	5	4
Jul 2021 - Jun 2021	Dummy Topic 2	6	3

Download certificate 

[Question review stats](#)

Mobile Views: [Mocks](#)

- The default view will be:
 - View -> Overview
 - Contribution Type -> Translation Contributions
 - Language -> Should be selected(Language selection will only include the languages that the contributor has translated.)



- Previous/Next text are added for the pagination purpose of the stats table.
- Reviewer views will not have "Download certificate" button since it is a future work.

[My Contributions](#)
[Submit Question](#)
[Translate Text](#)
[My Stats](#)

View:

Contribution Type:

Language:

Translation Contributions - Hindi

Months	Topic Names	Accepted Cards	Accepted Word Count
Jun 2021 - Jul 2021	Dummy Topic 1	7	150
Jun 2021 - Jul 2021	Dummy Topic 2	5	120
Jun 2021 - Jul 2021	Dummy Topic 3	5	120
Jun 2021 - Jul 2021	Dummy Topic 4	5	120
Jun 2021 - Jul 2021	Dummy Topic 5	5	120

[< Previous](#)
[Next >](#)

[Download certificate](#)

[My Contributions](#)
[Submit Question](#)
[Translate Text](#)
[My Stats](#)

View:

Contribution Type:

Language:

Translation Reviews - Hindi

Months	Topic Names	Reviewed Cards	Reviewed Word Count	Accepted Cards	Accepted Word Count
Jun 2021 - Jul 2021	Dummy Topic 1	7	150	5	120
Jun 2021 - Jul 2021	Dummy Topic 2	7	150	5	120


[My Contributions](#)
[Submit Question](#)
[Translate Text](#)
[My Stats](#)

View

Contribution Type

Question Contributions

Months	Topic Names	Accepted Questions	Accepted Questions Without Edits
Jun 2021 - Jul 2021	Dummy Topic 1	7	6
Jun 2021 - Jul 2021	Dummy Topic 2	5	4
Jun 2021 - Jul 2021	Dummy Topic 3	5	3
Jun 2021 - Jul 2021	Dummy Topic 4	5	2

[Download certificate](#) 

[My Contributions](#)
[Submit Question](#)
[Translate Text](#)
[My Stats](#)

View

Contribution Type

Question Reviews

Months	Topic Names	Reviewed Questions	Accepted Questions
Jun 2021 - Jul 2021	Dummy Topic 1	7	6
Jun 2021 - Jul 2021	Dummy Topic 2	5	4
Jun 2021 - Jul 2021	Dummy Topic 3	5	3
Jun 2021 - Jul 2021	Dummy Topic 4	5	2

My Contributions Submit Question Translate Text **My Stats**

View

Badge Type

Language

Submissions

1 Submission
Hindi

10 Submissions
Hindi

20 Submissions
Hindi

50 Submissions
Hindi

Acceptances

1 Acceptance
Hindi

10 Acceptances
Hindi

Reviews

1 Review
Hindi

10 Reviews
Hindi

Corrections

1 Corrections
Hindi

10 Corrections
Hindi

My Contributions Submit Question Translate Text **My Stats**

View

Badge Type

Submissions

1 Submission

10 Submissions

20 Submissions

Acceptances

1 Acceptance

10 Acceptances

Reviews

1 Review

10 Reviews

Corrections

1 Correction









10 Corrections

- Create a new component core/templates/pages/contributor-dashboard-page/stats-type-selector as the selector to choose what contribution stats needs to be fetched.
 - Inputs for the component
 - activeContributorStatsType: string
 - Outputs from the component
 - setActiveContributorStatsType: EventEmitter<string>
- Create a new component core/templates/pages/contributor-dashboard-page/contributor-badge component to generate contributor badges from the stats.
 - Inputs for the component
 - contributionType: String.
 - contributionSubType: String.
 - languageCode: String|None.
 - numberOfContributions: Integer.
 - numberOfContributionsRemaining: Integer.
 - badgeLocked: boolean.
 - Texts in the badge within tooltips
 - For translations
 - {numberOfContributions} {contributionType} {contributionSubType} in {language} language
 - 1 translation acceptance in Hindi language
 - 10 translation reviews in Hindi language
 - 1 translation correction in Hindi language
 - Achieve {numberOfContributions} {contributionType} {contributionSubType} in {language} language to unlock
 - Achieve 20 translation acceptances in Hindi language to unlock
 - For questions
 - {numberOfContributions} {contributionType} {contributionSubType}
 - 1 question acceptance
 - Achieve {numberOfContributions} {contributionType} {contributionSubType} to unlock
 - Achieve 20 question acceptances to unlock
 - Handling long language names

- The letter sizes will be made smaller considering the number of letters contained in the language name. This will also be considered with the width of the ribbon that the language text will be put.
 - 1 - 5 letters: 16pt
 - 6 - 10 letters: 15 pt
 - 11 - 15 letters: 14 pt
- Font size can be set dynamically like the following:

```
defaultFontSize = 16
fontSize = defaultFontSize - (1 * Math.floor(languageName.length / 5))
```

- **Note: need to test with all the different language names in the contributor dashboard when building the badges UI.**

For Translations		For Questions	
			
			

Documentation changes

N/A

Testing Plan

E2e testing plan

General Setup

- create an Curriculum Admin user
- populate opportunities
 - create topic
 - create skill and assign topic
 - create skill with description and explanation
- create 4 different users and assign them following roles
 - user1: translation submitter
 - user2: translation reviewer
 - user3: question submitter
 - user4: question reviewer
- login as translation submitter
 - submit translations
- logout translation submitter
- login as translation reviewer
 - review and accept/reject translations
- logout translation reviewer
- login as question submitter
 - submit questions
- logout question submitter
- login as question reviewer

- review and accept/reject questions
- logout question reviewer

#	Test name	Initial setup step	Step	Expectation
1.	Translation Contributor stats	Login as a translator	Goto contributor dashboard page	The page loads
			Click on "My Stats" tab and select "Overview" from the left side menu.	My stats section loads
			Click on the "Translation Contributions" in the contributor stats selector	Selected contributor stats type appears in the selector and the stats table loads.
			Click on "Download" button	Contributor certificate is available
2.	Translation Contributor badges		Select "Badges" from the left side menu.	Badges section loads
				Translation contributor badges available in the page.
3.	Translation Reviewer stats	Login as a translation reviewer	Goto contributor dashboard page	The page loads
			Click on "My Stats" tab and select "Overview" from the left side menu.	My stats section loads
			Click on the "Translation Reviews" in the contributor stats selector	Selected contributor stats type appears in the selector and the stats table loads.
4.	Translation Reviewer badges		Select "Badges" from the left side menu.	Badges section loads
				Translation reviewer badges available in the page.
5.	Question Contributor stats	Login as a question submitter	Goto contributor dashboard page	The page loads
			Click on "My Stats" tab	My stats section loads

			Click on the "Question Contributions" in the contributor stats selector	Selected contributor stats type appears in the selector and the stats table loads.
			Click on "Download" button	Contributor certificate is available
6.	Question Contributor badges		Select "Badges" from the left side menu.	Badges section loads
				Question contributor badges available in the page.
7.	Question Reviewer stats	Login as a question reviewer	Goto contributor dashboard page	The page loads
			Click on "My Stats" tab	My stats section loads
			Click on the "Question Reviews" in the contributor stats selector	Selected contributor stats type appears in the selector and the stats table loads.
4.	Question Reviewer badges		Select "Badges" from the left side menu.	Badges section loads
				Question reviewer badges available in the page.

Feature testing

Does this feature include non-trivial user-facing changes?

YES

- E2e tests will be present for the features

Implementation Plan

Milestone 1(16 June - 5 Sept)

Objectives

- Creating all the storage models and initializing them

- Creating all the required backend services
- Creating/updating all the required controllers
- Writing all the backend tests required for the project

No.	Description of PR / action	Prereq PR numbers	Target date for PR creation	Target date for PR to be merged
1	Introduce storage models for contribution stats, latest milestone info	-	16/6	21/6
2	Introduce domain objects for contribution stats, contributor milestone email info, contributor badge info	-	13/7	18/7
3	Add required suggestion services to handle contributor stats	1, 2	27/7	1/8
4	Add required email services to notify contributors about milestones they achieved	1, 2	6/8	11/8
5	Write asynchronous tasks to send milestone notifying emails	1, 2, 3, 4	18/8	25/8
6	Implement controller layer for the project	3	28/8	1/9
5 Days of revisions/bug fixes				

Milestone 2(Sep 13 - Nov 13)

Objectives

- Adding all frontend services
- Creating contributor stats component
- Implement contributor badge creation in frontend
- Implement contributor certificate generation

No.	Description of PR / action	Prereq PR numbers	Target date for PR creation	Target date for PR to be merged
-----	----------------------------	-------------------	-----------------------------	---------------------------------

7	Add new frontend endpoints to call the ContributorStatsSummariesHandler	6	13/9	18/9
8	Create contributor stats component to contributor dashboard with mobile view	7	30/9	6/10
9	Add contributor badges to stats page with the mobile view	9	12/10	16/10
10	Add certificate generation for contributors (including service method)	8	22/10	26/11
11	Cover e2e tests for contributor stats and contributor badges	7, 8, 11	6/11	10/11

Future Work

Note: This section is mainly for reference (since it is understood that items in this section will not be part of the GSoC project). Proposals will primarily be evaluated based on the implementation plan above.

- Reviewer certificate generation will be a future work.
- Contributor certificate verification via QR codes will be a future work. The following changes will be required for this.

Controllers

ContributorDashboardCertificateVerificationPage

- A new endpoint that will load the contributor dashboard certificate verification page. This will be called when the QR code in the certificate is scanned.
- URL: `./contributor-dashboard-certificate-verification/<username>/<contribution_stats_type>`
- GET:
 - *Decorator:* @open_access
 - *Params:*
 - `contribution_stats_type`: str. The type of the stats summary needs to be fetched. This can be either of `translation_contributions`, `translation_reviews`, `question_contributions`, `question_reviews`.
 - `contributor_username`: str.

- *Data Response:* html page
 - Returns the contributor-dashboard-accomplishments-verification-page with contributor_username and contribution_stats_type.
- *Functionality:*
 - Load the contributor-dashboard-accomplishments-verification-page

Web frontend changes

- Create a new page core/templates/pages/contributor-dashboard-certificate-verification-page to verify contributor accomplishments.
 - This page will be opened when someone scans the QR code which is attached to the certificates of contributors. This will internally call the ContributorStatsSummariesHandler to load and verify the contributors' stats summary.