

About You

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

Oppia is such a large community with contributors around all over the world and its mission to “help anyone learn anything they want in an effective way” urges me to contribute more towards its growth. I am also interested in contributing to this vision so that everyone across the world has access to high-quality scalable lessons. I learned so much in just a few months working with Oppia and I would continue to contribute to this wonderful community even after the GSoC period ends. The mentors are really helpful and they helped me improve my technical and coding skills. Contributing to Oppia taught me a lot about Open Source and Project management. So I would like to take this opportunity to develop something significant for the organisation. I started with Oppia under the event called DSC-SLoP which made me kickstart my journey to Open Source.

What interests me about this project?

It was fun working with Angular framework under the Migration Team over the past months and I think the project would give better insights for the same.

The developers play an important role in the growth of the site, I believe that after completion of this project they would have a smooth experience to contribute even more to the community.

Prior experience

I have been contributing to oppia for the last 5-6 months and I am familiar with most parts of the codebase. I have been working with Angular 8+ and AngularJS for the last 9-10 months now. I have also been a part of release testing for the past 3+ months. I am also involved as a backend developer in my university Fests and other events. I have been involved with web development for the last two years.

These are some of the projects I have created:

- [Angular Blog\(MEAN\)](#)-Users can add, delete and edit the posts uploaded, used Mongo-DB Atlas for the backend. Also added auth-guards, angular-routing in the same
- [Recipe Website in Angular](#)-Users can order groceries and food items of their choice and add it in their cart for checkout purposes.
- [Covitrax_](#) Users can track the patient count by searching up any area they are living in. Added graph support using Canvas.js and authentication using firebase.

I strongly believe that these projects along with my experience with Oppia for the past six months should help me continue my GsoC journey smoothly.

Link to previous PR's

- [Migrating subtopic-viewer-navbar-pre-logo-action.component.ts\(#11983\)](#)
- [Migrating Collection Local Nav to Angular\(#11352\)](#)

- Make Typescript checks strict for some services(#[12561](#))
- Make Typescript Checks Strict for CreatorDashboardStatsObjectFactory(#[11332](#))
- Make Type Checks Strict for CreatorExplorationSummaryObjectFactory(#[11327](#))
- Make Typescript Checks strict for some services(#[12569](#))
- Migrate Create-Activity-Button to Angular (#[12290](#))

Link to all my contributions to oppia is ([here](#))

Contact info and timezone(s)

- Email: mridulsetia18@gmail.com
- Phone: +919899160754
- Github: [mridul-netizen](#)

I will stay in India throughout the summer. The time zone will be Indian Standard Time (GMT+5:30)

Time commitment

- I would be working on the GSoC project throughout the 10-week period, from 7th June to 16th August.
- I would commit at least 3-4 hrs per day during the coding period, or more if required.
- I would have my exams from the second week of July so I would give 1-2 hours during that period and would cover that by working for extra hours on the weekends.

S.No	Dates	Days(Total)	Time Commitment
1.	7th June - 13th June	Mon - Sun(7)	3-4h/day - 20h/week
2.	14th June - 20th June	Mon - Sun(7)	3-4h/day - 20h/week
3.	21st June - 27th June	Mon - Sun(7)	3-4h/day - 20h/week
4.	28th June - 4th July	Mon - Sun(7)	3-4h/day - 20h/week
5.	5th July - 11th July	Mon - Sun(7)	3-4h/day - 20h/week
6.	12th July - 18 July	Mon - Sun(7)	3-4h/day - 20h/week
7.	19 July - 25 July	Mon - Sun(7)	1-2h/day - 10h/week
8.	26 July - 1 August	Mon - Sun(7)	3-4h/day - 22h/week
9.	2 August - 8 August	Mon - Sun(7)	3-4h/day - 22h/week
10.	8 August - 14 August	Mon - Sun(7)	3-4h/day - 22h/week

Essential Prerequisites

Answer the following questions:

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

```
Done in 0.96s.
Installing pre-commit hook for git
Symlink already exists
Making pre-commit hook file executable ...
pre-commit hook file is now executable!
Installing pre-push hook for git
Symlink already exists
Making pre-push hook file executable ...
pre-push hook file is now executable!
-----
Tasks still running:
  core.controllers.admin_test (started 13:18:42)
-----
07:49:40 FINISHED core.controllers.admin_test: 57.7 secs

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

SUCCESS core.controllers.admin_test: 108 tests (49.2 secs)

Ran 108 tests in 1 test class.
All tests passed.

Done!
mridul@linux-mridul:~/Desktop/opensource/oppia$
```

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

```
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4258 of 4267 SUCCESS (0 secs / 49.112 secs)
LOG: 'Spec: Collection update service should create a proper backend change dict for changing categories has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4259 of 4267 SUCCESS (0 secs / 49.116 secs)
LOG: 'Spec: Collection update service should set/unset changes to a collection's category has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4260 of 4267 SUCCESS (0 secs / 49.119 secs)
LOG: 'Spec: Collection update service should set/unset changes to a collection's tags has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4261 of 4267 SUCCESS (0 secs / 49.122 secs)
LOG: 'Spec: Error page should check if status code is a number has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4262 of 4267 SUCCESS (0 secs / 49.126 secs)
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4263 of 4267 SUCCESS (0 secs / 49.149 secs)
LOG: 'Spec: Error page should set images and page title when $onInit triggers has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4263 of 4267 SUCCESS (0 secs / 49.149 secs)
LOG: 'Spec: Error page should set images and page title when $onInit triggers has passed'
LOG: 'Spec: Question Player Concept Card Modal Controller should initialize $scope properties after controller is initialized has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4264 of 4267 SUCCESS (0 secs / 49.175 secs)
LOG: 'Spec: Question Player Concept Card Modal Controller should go to next concept card, and identify when it is the last concept card. has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4265 of 4267 SUCCESS (0 secs / 49.184 secs)
LOG: 'Spec: Question Player Concept Card Modal Controller should refresh page when retrying a practice test has passed'
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4266 of 4267 SUCCESS (0 secs / 49.194 secs)
Chrome Headless 89.0.4389.90 (Linux x86_64): Executed 4267 of 4267 SUCCESS (56.724 secs / 49.202 secs)
TOTAL: 4267 SUCCESS
TOTAL: 4267 SUCCESS
15 03 2021 13:15:32.443:WARN [launcher]: ChromeHeadless was not killed in 2000 ms, sending SIGKILL.
Done!
mridul@linux-mridul:~/Desktop/opensource/oppia$
```

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

```
7 specs, 0 failures
Finished in 194.474 seconds

Executed 7 of 7 specs SUCCESS in 3 mins 14 secs.
[13:38:54] I/launcher - 0 instance(s) of WebDriver still running
[13:38:54] I/launcher - chrome #01 passed

t emulators: Received SIGTERM for the first time. Starting a clean shutdown.
t emulators: Please wait for a clean shutdown or send the SIGTERM signal again to stop right now.
t emulators: shutting down emulators.
t ut: Stopping Emulator UI
t Emulator UI has exited upon receiving signal: SIGTERM
t Authentication Emulator
t LibreOfficeWriter emulator hub
t Logging: Stopping Logging Emulator
Traceback (most recent call last):
  File "/usr/lib/python2.7/runpy.py", line 174, in _run_module_as_main
    _main(_name, loader, pkg_name)
  File "/usr/lib/python2.7/runpy.py", line 72, in _run_code
    exec code in run_globals
  File "/home/mridul/Desktop/opensource/oppla/scripts/run_e2e_tests.py", line 618, in <module>
    main()
  File "/home/mridul/Desktop/opensource/oppla/scripts/run_e2e_tests.py", line 599, in main
    flake_checker.report_pass(parsed_args.suite)
  File "scripts/flake_checker.py", line 116, in report_pass
    metadata = _get_build_info()
  File "scripts/flake_checker.py", line 111, in _get_build_info
    raise Exception('Unknown build environment.')
Exception: Unknown build environment.
Killing java -Djava.security.egd=file:///dev/.urandom -Dwebdriver.chrome.driver=/home/mridul/Desktop/opensource/oppla/node_modules/webdriver-manager/selenium/chromedriver_89.0.4389.23 -Dwebdriver.gecko.driver=/home/mridul/Desktop/opensource/oppla/node_modules/webdriver-manager/selenium/geckodriver-v0.29.0 -jar /home/mridul/Desktop/opensource/oppla/node_modules/webdriver-manager/selenium/selenium-server-standalone-3.141.59.jar -port 4444 ...
[13:38:55] I/start - Selenium Standalone has exited with code null
Cleaning up the redis servers.
mridul@192.168.1.101:~/Desktop/opensource/oppla$
```

Other summer obligations

I will be having my end-semester examinations in the second week of July. I can cover that up in the vacations after my exams. I don't have any other obligations.

Communication channels

Oppia members mainly use google-hangouts, discord and gitter to discuss workflow to be followed ahead. I too favor communication through any of them. I am also comfortable with e-mails.

Application to multiple orgs

I'm only applying for oppia this year, however I'm working on yet another proposal (Angular Migration).

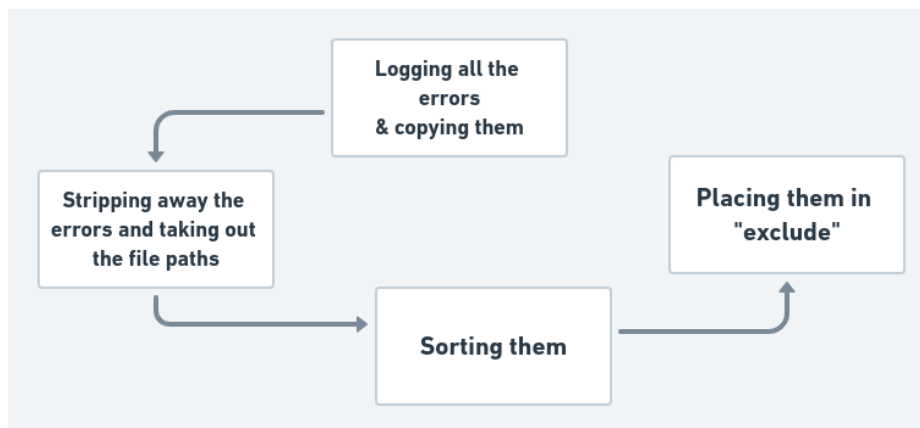
With completion of this project strict typings will be introduced to approximately all of the upgraded services along with their tests.

Technical Design

Implementation Approach



Currently the tsconfig-strict.json is used only with our `typescript_checks.py`. This file is used during pushing new changes hence in `pre_push_hook.py`. The services being singleton in nature and being dependent on the objectFactories/model files need to be covered first. The current implementation requires listing of files which are currently passing strict type checks. This implementation can be improved by instead excluding the files for which we don't want to check the strict type, much like the frontend tests using `NOT_FULLY_COVERED_FILENAMES`. Starting with the files that are not covered with strict checks, I created a list of them [here](#).



Process for creating such a list

This list includes every file that ends with a *.ts extension excluding the files that are already

present in the files property in our ["tsconfig-strict.json"](#).

My approach involves covering directories along with files. I will be doing that by only using "exclude" and "include" properties. The **"exclude"** property will list all the files that are not strictly typed as covered in the spreadsheet above and the **"include"** will cover "files as soon as they will be strictly typed + the already covered files present currently in tsconfig-strict.json". To be more precise and clear let's take an example.

Since the project demands to work on files from [UpgradedServices.ts](#) suppose I am

- working on `core/templates/pages/collection-editor-page/services/collection-editor-state.service.ts` .I will remove this file from exclude and put that in include.
- I will run `python -m scripts.typescript_checks --strict_checks` This will give me some information regarding errors, if the file is not strictly typed.
- I will fix them and move to the next file.

Conclusion

The lines from "exclude" will decrease and in "include" would increase. Suppose all services inside `core/templates/pages/collection-editor-page/services/` are now strictly typed then we can simply remove all those filenames from include and put a path instead i.e

`core/templates/pages/collection-editor-page/services/*.ts` .

(Note: We can't directly put paths in include and simply exclude files because typescript gives more preference to "include" over "exclude" .Checks would simply give errors in all those files which are excluded.)

Updated tsconfig-strict.json would look like:

```
"include": [  
  "Already_covered_files",  
  "Newly_typed_Files",  
  "typings/*.ts"  
],  
"exclude": [  
  "Not_covered_files"  
]
```

Why would "exclude" be useful?

To keep a track of files that are not fully covered.

Approach for Resolving Errors:

I analyzed the upgraded services, made some PRs for this issue and also looked into existing PRs and found causes of errors along with their appropriate solutions.

1. Null assignments to defined variables

Null assignments can be removed from the places where not required. This could also help us reduce bugs. One such bug which I got due to this issue was

```
✖ ▶ Every parameter passed into interpolateUrl must have string values, but received: {filename: null} logger.service.ts:60
✖ ▶ ERROR Error: Uncaught (in promise): Error: Every parameter passed into interpolateUrl must have string values, but received: {filename: null} core.js:4868
Error: Every parameter passed into interpolateUrl must have string values, but received: {filename: null}
    at AlertsService.fatalWarning (alerts.service.ts:78)
    at UrlInterpolationService.interpolateUrl (url-interpolation.service.ts:137)
    at AssetsBackendApiService.getDownloadUrl (assets-backend-api.service.ts:151)
    at AssetsBackendApiService.getThumbnailUrlForPreview (assets-backend-api.service.ts:148)
    at StoryViewerPageComponent.generatePathIconParameters (story-viewer-page.component.ts:85)
    at eval (story-viewer-page.component.ts:145)
    at ZoneDelegate.invoke (zone.js:390)
    at Object.onInvoke (core.js:27960)
    at ZoneDelegate.invoke (zone.js:389)
    at Zone.run (zone.js:147)
    at resolvePromise (zone.js:836)
    at eval (zone.js:902)
    at ZoneDelegate.invokeTask (zone.js:425)
    at Object.onInvokeTask (core.js:27948)
    at ZoneDelegate.invokeTask (zone.js:424)
    at Zone.runTask (zone.js:192)
    at drainMicroTaskQueue (zone.js:605)
    at ZoneTask.invokeTask [as invoke] (zone.js:511)
    at invokeTask (zone.js:1675)
    at XMLHttpRequest.globalZoneAwareCallback (zone.js:1712)
```

This can be resolved with the following change at appropriate position

```
providedIn: 'root'
})
export class AngularNameService {
-   static angularName: string = null;
+   static angularName: string;
}
getNameOfInteractionRulesService(inter
```

Unlike the above change there can be a case where we have to add '| null' or '| undefined' to the existing type. The error noticed related to this is mentioned below.

2. Argument of type 'null' is not assignable to parameter of type 'string[]'

This happens when we intend to return null from a function in some 'if-else' condition. Either we can return something like [' '] or we can change the return type to **string [] | null**. Changing return statements can break the code at places where function is called therefore this would be changed accordingly.


```

getExplorationTitle(): string {
  if (this._explorationSummaryObject) {
getExplorationTitle(): string | null {
  if (this.doesExplorationExist()) {
    return this._explorationSummaryObject.title;
  } else {
    return null;
  }
}

```

3. No index signature with parameter 'string' found on type { }

We can fix these errors by validating our input. It might be the case that the value we are assigning to a variable is of some different type. The inferred types are not exact in some cases. When inferring variable types, TypeScript takes into account only the line where the variable is declared. Sometimes it infers { } for object types. So we will need to explicitly type them for such cases. Here I will try to define an interface if needed for the particular type and assign it to the variable or constant.

```

+ interface NodeTitles {
+   [key: string]: string
+ }
+

```

```

const nodeToMove = this._nodes[fromIndex];
const nodeToMove: StoryNode = this._nodes[fromIndex];

```

4. Being sure in case that all expected parameters are passed to a target

Error:-Type {... } is not assignable to type EmailDashBoardQueryDict

If the particular type has not been defined for a target then it is difficult to infer type for the parameters we're passing to it. In such a case first the target needs to be typed correctly and then correct parameters are passed.

```

var recentQueries = [
var recentQueriesDict: EmailDashboardQueryDict[] = [
{
  query: {
    id: 'q123',
    status: 'processing'
  }
},
  id: 'q123',
  status: 'processing',
  num_qualified_users: 0,
  submitter_username: '',
  created_on: ''
},
{
  query: {
    id: 'q456',
    status: 'processing'
  }
}
].map(EmailDashboardQuery.createFromBackendDict);
  id: 'q456',
  status: 'processing',
  num_qualified_users: 0,
  submitter_username: '',
  created_on: ''
}];
var recentQueries = recentQueriesDict.map(
  EmailDashboardQuery.createFromQueryDict);

```

5. .map for Object.keys

Here the **Error:-Argument of type 'string[]' is not assignable to parameter of type 'number[]'** specifies that we could not assign a wrong type to some variable of type string[]. Since we get the data in number[] format we would map it as shown.

```

var numSpaces = line.length - this.removeLeadingWhitespace(line).length;

var existingNumSpaces = Object.keys(numSpacesToDesiredIndentLevel);
var existingNumSpaces = Object.keys(
  numSpacesToDesiredIndentLevel
).map(Number);
var maxNumSpaces = Math.max.apply(null, existingNumSpaces);
if (numSpaces > maxNumSpaces) {
  // Add a new indentation level.
}
}
}

@@ export class CodeNormalizerService {
}

if (isShortfallLine) {
  existingNumSpaces = Object.keys(numSpacesToDesiredIndentLevel);
  existingNumSpaces = Object.keys(
    numSpacesToDesiredIndentLevel
  ).map(Number);
  numSpaces = Math.max.apply(null, existingNumSpaces);
}
}
}

```

Some other errors that were found:-

6. Lacks ending return statement and return type not include undefined

Will add a return statement and the appropriate return type wherever possible.
To understand the cause, let us look at this example:

```

foo(a: Interface, b: Interface): boolean {
  for (let i = 0; i < aProps.length; i++) {
    let propName = aProps[i];
    if (!this.foo(a[propName], b[propName])) {
      return false;
    }
  }
}
}
}

```

The error was caused due to not considering all cases of this.foo()'s return value.

Only the false return value of the function is covered here.

This will be fixed by covering all possible cases or add a default return value to the function

Covering all cases:

```
foo(a: Interface, b: Interface): boolean {
  for (var i = 0; i < aProps.length; i++) {
    var propName = aProps[i];
    if (!this.foo(a[propName], b[propName])) {
      return false;
    } else {
      return true;
    }
  }
}
```

7. Variable is used before assignment

Will check the control flow logic and make sure the variable has a valid value before control passes to any statement that reads it.

8. Variable implicitly has a type 'any'

Will resolve this by adding appropriate types.

9. Parameter implicitly has any type

This is caused when there is no type specified for the function parameters or any variable. I will explicitly find out the type here. Some solutions to this problem would be to

1. Define appropriate interfaces if needed.
2. Determining the type by analyzing the existing and imported files.
3. Determining types by analyzing the return statements.
4. Console logging the value of variables.

10. Type 'null' cannot be used as an index type.

TypeScript won't allow for null or undefined as keys in objects. **TypeScript** only allows two **types** for **indexes** (the keys) i.e. string and number. I will fix this by assigning proper values in place of null .

There could also be some cases where removing null could break the existing code, so we can also check for the index to be not null.

Cause: This error occurs when a variable holding null value is used to index an array.

Eg.

```

let array: number[] = [1, 2, 3];
let index: number | null = null;
let b: number;

foo(index: number): void {
    this.b = arr[index];
}

```

The fix for this would be to check for the value of the index variable to not be null before usage.

```

let array: number[] = [1, 2, 3];
let index: number = null;
let b: number;

foo(index: number): void {
    if (index !== null) {
        this.b = arr[index];
    }
}

```

11. Object is possibly 'undefined'.

Instantiation/ Initialization of objects could help resolve these types of errors

12. There are some places in the codebase where null and undefined is not needed at all and the existing code(like the tests)passes depending on them.Eg.

```

exploration_summary: null
exploration_summary: {
  category: '',
  community_owned: false,
  activity_type: '',
  last_updated_msec: 0,
  ratings: {
    1: 0,
    2: 0,
    3: 0,
    4: 0,
    5: 0,
  },
  id: '',
  created_on_msec: 0,
  human_readable_contributors_summary: {},
  language_code: '',
  num_views: 0,
  objective: '',
  status: '',
  tags: [''],
  thumbnail_bg_color: '',
  thumbnail_icon_url: '',
  title: '',
}

```

Here the property `exploration_summary` was defined `null` previously, it is of type `LearnerExplorationSummaryBackendDict` so I defined it with some empty values as shown. Here `exploration_summary` could not have been defined with type `LearnerExplorationSummaryBackendDict | null` because of one method which demands non-null values to be passed with this property.

```

expect(summaryObject).toBeNull();
expect(summaryObject).not.toBeNull();
expect(collectionNode.getExplorationTitle()).toBeNull();
expect(collectionNode.isExplorationPrivate()).toBeUndefined();
expect(collectionNode.isExplorationPrivate()).toBeFalse();

```

This made these tests to fail so these have to be refactored accordingly to pass.

```
// function is undefined if doesExplorationExist() returns false.
// function is false if doesExplorationExist() returns false.
isExplorationPrivate(): boolean {
  if (this._explorationSummaryObject) {
    if (this.doesExplorationExist()) {
      return this._explorationSummaryObject.status === (
        AppConstants.ACTIVITY_STATUS_PRIVATE);
    } else {
      return undefined;
    }
    return false;
  }

  doesExplorationExist(): boolean {
    return this._explorationSummaryObject !== null;
    return this._explorationSummaryObject.title !== '';
  }
}
```

Also there were some changes in the logic of how explorationSummaryObject was used to check for conditions, so this approach was changed too. Here the property title is must and is required for an exploration to exist or published, so we are checking if it is not empty thus returning same boolean logic as before. Through this approach we can minimize the use of null where it is not needed at all.

Testing Approach

Any changes made to the services will be covered by unit tests. Type checking would be also done on respective spec files.

Milestones

Looking at the places where we can find services

Directories	No. Of Files
core/templates/pages/**/*service.ts	89 files

core/templates/services/	104 files
core/templates/domain/	55 files
core/templates/components/	24 files
core/templates/expressions/	4 files
extensions/	52 files

O R D E R	Names Of Services	Dependencies on Other Files
0.	<ol style="list-style-type: none"> 1. AlgebraicExpressionInputRulesService 2. AppService 3. AngularNameService 4. AudioBarStatusService 5. AudioTranslationManagerService 6. AutoplayedVideosService 7. BackgroundMaskService 8. baseInteractionValidationService 9. BrowserCheckerService 10. CodeNormalizerService 11. CollectionValidationService 12. ComputeGraphService 13. ConstructTranslationIdsService 14. ContinueRulesService 15. CountVectorizerService 16. CsrftokenService 17. DateTimeFormatService 18. DebouncerService 19. DragAndDropSortInputRulesService 20. EditabilityService 21. EndExplorationRulesService 22. EndExplorationRulesService 23. ExplorationDiffService 24. ExplorationFeaturesService 25. ExpressionParserService 26. ExternalSaveService 27. GraphDetailService 28. GraphUtilsService 	NA

	<p>29. I18nLanguageCodeService 30. IdGenerationService 31. ImageClickInputRulesService 32. ImprovementsService 33. InteractionDetailsCacheService 34. InteractiveMapRulesService 35. InteractionSpecsService 36. ItemSelectionInputRulesService 37. LoaderService 38. LocalStorageService 39. LoggerService 40. LogicProofRulesService 41. MathEquationInputRulesService 42. MultipleChoiceInputRulesService 43. MusicPhrasePlayerService 44. NumberAttemptsService 45. NumericExpressionInputRulesService 46. NumericInputRulesService 47. PlayerCorrectnessFeedbackEnabledService 48. RatingComputationService 49. RatioExpressionInputRulesService 50. ReviewTestEngineService 51. SetInputRulesService 52. SolutionValidityService 53. StateGraphLayoutService 54. SuggestionModalService 55. SuggestionsService 56. ThreadStatusDisplayService 57. TopicsAndSkillsDashboardPageService 58. UtilsService 59. VersionTreeService 60. WinnowingPreprocessingService 61. baseInteractionValidationService</p>	
<p>1.</p>	<p>1. AlgebraicExpressionInputValidationService 2. AlertsService 3. CodeReplValidationService 4. ContinueValidationService 5. DeviceInfoService 6. DocumentAttributeCustomizationService 7. DragAndDropSortInputValidationService 8. EndExplorationValidationService 9. ExpressionSyntaxTreeService 10. FractionInputRulesService 11. FractionInputValidationService 12. GraphInputRulesService 13. GraphInputValidationService</p>	<ul style="list-style-type: none"> ● baseInteractionValidationService ● LoggerService ● WindowRef ● baseInteractionValidationService ● ExpressionParserService ● FractionObjectFactory ● UtilsService ● GraphUtilsService ● GuestCollectionProgressService ● FractionObjectFactory ● UnitsObjectFactory ● SubtitledHtmlObjectFactory ● ParamChangesObjectFactory

	<ol style="list-style-type: none"> 14. HtmlEscaperService 15. ImageClickInputValidationService 16. InteractiveMapValidationService 17. ItemSelectionInputValidationService 18. LogicProofValidationService 19. MathEquationInputValidationService 20. MessengerService 21. MetaTagCustomizationService 22. MultipleChoiceInputValidationService 23. MusicNotesInputRulesService 24. MusicNotesInputValidationService 25. NormalizeWhitespacePipe 26. NumericInputValidationService 27. NumericExpressionInputValidationService 28. NumberWithUnitsRulesService 29. PageTitleService 30. PencilCodeEditorValidationService 31. PlayerTranscriptService 32. RatioExpressionInputValidationService 33. SetInputValidationService 34. SkillCreationBackendApiService 35. SpeechSynthesisChunkerService 36. SVMPredictionService 37. SchemaDefaultValueService 38. StateClassifierMappingService 39. SolutionValidityService 40. TextInputValidationService 41. TopicCreationBackendApiService 42. WindowDimensionsService 	<ul style="list-style-type: none"> ● ParamTypeObjectFactory ● RecordedVoiceoversObjectFactory ● VoiceoverObjectFactory ● SubtitledHtmlObjectFactory ● SubtitledUnicodeObjectFactory ● StoryContentsObjectFactory
<ol style="list-style-type: none"> 2. 	<ol style="list-style-type: none"> 1. CkEditorCopyContentService 2. AutogeneratedAudioPlayerService 3. BottomNavbarStatusService 4. PreventPageUnloadEventService 5. CodeReplPredictionService 6. CodeReplRulesService 7. ContextService 8. EditorFirstTimeEventsService 9. ExplorationImprovementsTaskRegistryService 10. ExpressionEvaluatorService 11. ExtensionTagAssemblerService 12. FocusManagerService 13. IdGenerationService 14. LanguageUtilService 15. NumberWithUnitsValidationService 16. PencilCodeEditorRulesService 	<ul style="list-style-type: none"> ● AnswerGroupObjectFactory ● OutcomeObjectFactory ● RuleObjectFactory ● HtmlEscaperService ● SpeechSynthesisChunkerService ● WindowDimensionsService ● CountVectorizerService ● SVMPredictionService ● SubtitledHtmlObjectFactory ● WorkedExampleObjectFactory ● RecordedVoiceoversObjectFactory ● ConceptCardObjectFactory ● ExpressionParserService ● ExpressionSyntaxTreeService ● BrowserCheckerService ● NumberWithUnitsObjectFactory ● baseInteractionValidationService

	<ol style="list-style-type: none"> 17. SidebarStatusService 18. StateContentService 19. StateCustomizationArgsService 20. StateInteractionIdService 21. StateNextContentIdIndexService 22. StatePropertyService 23. StateRecordedVoiceoversService 24. StateSolicitAnswerDetailsService 25. StateSolutionService 26. StateWrittenTranslationsService 27. TextInputRulesService 28. StoryEditorNavigationService 29. UrlInterpolationService 30. ValidatorsService 	<ul style="list-style-type: none"> ● ParamSpecObjectFactory ● NormalizeWhitespacePipe ● CodeNormalizerService ● NormalizeWhitespacePunctuationAndCasePipe ● CodeNormalizerService ● AlertsService ● RecordedVoiceoversObjectFactory ● SubtitledHtmlObjectFactory ● SubtopicPageContentsObjectFactory ● SuggestionThreadObjectFactory ● StoryContentsObjectFactory ● ShortSkillSummaryObjectFactory ● StoryReferenceObjectFactory ● StoryReferenceObjectFactory
<ol style="list-style-type: none"> 3. 	<ol style="list-style-type: none"> 1. AdminBackendApiService 2. AdminDataService 3. AssetsBackendApiService 4. EmailDashboardBackendApiService 5. ExplorationPermissionsBackendApiService 6. InteractionRulesRegistryService 7. AudioTranslationLanguageService 8. ConceptCardBackendApiService 9. ClassroomBackendApiService 10. CollectionCreationBackendService 11. CollectionRightsBackendApiService 12. ContributionOpportunitiesBackendApiService 13. CreatorDashboardBackendApiService 14. CurrentInteractionService 15. ExplorationFeaturesBackendApiService 16. ExplorationFeaturesBackendApiService 17. ExplorationHtmlFormatterService 18. ExplorationImprovementsBackendApiService 19. ExplorationStatsBackendApiService 20. GenerateContentIdService 21. LearnerAnswerDetailsBackendApiService 22. LearnerDashboardBackendApiService 23. LearnerDashboardIdsBackendApiService 24. PlatformFeatureBackendApiService 25. PlatformFeatureAdminBackendApiService 26. PlatformFeatureDummyBackendApiService 27. PlayerPositionService 28. PlaythroughBackendApiService 	<ul style="list-style-type: none"> ● UrlInterpolationService ● CsrfTokenService ● ContextService ● AlgebraicExpressionInputRulesService ● CodeReplRulesService ● ContinueRulesService ● DragAndDropSortInputRulesService ● EndExplorationRulesService ● FractionInputRulesService ● GraphInputRulesService ● ImageClickInputRulesService ● InteractiveMapRulesService ● ItemSelectionInputRulesService ● LogicProofRulesService ● MathEquationInputRulesService ● MultipleChoiceInputRulesService ● MusicNotesInputRulesService ● NumberWithUnitsRulesService ● NumericExpressionInputRulesService ● NumericInputRulesService ● PencilCodeEditorRulesService ● RatioExpressionInputRulesService ● TextInputRulesService ● SetInputRulesService ● PlayerTranscriptService ● PlayerPositionService ● ExtensionTagAssemblerService

	<p>29. PlaythroughIssuesBackendApiService 30. PretestQuestionBackendApiService 31. ProfilePageBackendApiService 32. QuestionBackendApiService 33. ReadOnlyCollectionBackendApiService 34. SolutionVerificationService 35. SkillCreationBackendApiService 36. SkillMasteryBackendApiService 37. SkillRightsBackendApiService 38. StateParamChangesService 39. StateTopAnswersStatsBackendApiService 40. StatsReportingBackendApiService 41. StoryViewerBackendApiService 42. StoryViewerBackendApiService 43. TextInputPredictionService 44. TopicCreationBackendApiService 45. TopicsAndSkillsDashboardBackendApiService 46. TopicViewerBackendApiService 47. TranslationsBackendApiService 48. UserBackendApiService 49. UserService</p>	<ul style="list-style-type: none"> ● HtmlEscaperService ● StateNextContentIdIndexService ● LearnerDashboardIdsBackendApiService
<p>> 4.</p>	<p>1. CollectionCreationService 2. EditableCollectionBackendApiService 3. ExplorationRecommendationsService 4. ExtensionTagAssemblerService 5. PlatformFeatureService 6. PopulateRuleContentIdsService 7. PredictionAlgorithmRegistryService 8. ProfileLinkImageBackendApiService 9. UserExplorationPermissionsService 10. SubtopicViewerBackendApiService 11. AnswerClassificationService 12. AudioPreloaderService 13. ExplorationHtmlFormatterService 14. SubtopicViewerBackendApiService 15. StateInteractionStatsService 16. StateTopAnswersStatsService 17. InteractionAttributesExtractorService</p>	<ul style="list-style-type: none"> ● CollectionCreationBackendService ● AlertsService ● SiteAnalyticsService ● UrlInterpolationService ● LoaderService ● WindowRef ● ReadOnlyCollectionBackendApiService ● ContextService ● UrlService ● ExplorationRecommendationsBackendApiService ● ExplorationStatsBackendApiService ● HtmlEscaperService ● BrowserCheckerService ● TextInputPredictionService ● PredictionAlgorithmRegistryService ● CodeReplPredictionService ● SubtitledUnicodeObjectFactory ● SubtopicPageContentsObjectFactory ● RecordedVoiceoversObjectFactory

		<ul style="list-style-type: none"> ● SubtitledHtmlObjectFactory ● ReadOnlySubtopicPageObjectFactory ● StateCardObjectFactory ● AudioTranslationLanguageService ● AppService ● ReadOnlySubtopicPageObjectFactory ● ContextService ● SolutionObjectFactory ● LoggerService ● StateClassifierMappingService ● StatesObjectFactory ● ParamChangesObjectFactory ● InteractionSpecsService ● SubtopicPageContentsObjectFactory
--	--	--

There are currently 205 services in UpgradedServices.ts file that need to be fully covered. 39 of them are already covered leaving 166 files to be covered more. The project suggests to cover 120 services along with their spec so 240 in total. Since there are two milestones I would be covering around 55 files along with their tests so a total of 110 files for the first milestone and for the second milestone I would cover the remaining 65 making a total of 130 for the second milestone. The services also have some dependency on model.ts and ObjectFactories, so I will also cover them. Also if I got any extra time before the conclusion of GSoC I will try to cover all of the services left. If some of the services are already covered or there aren't any changes needed to be made within (just adding a path is enough) or some minor changes are there then I would not count them towards my goal and move to other files.

Milestone 1

Key Objective:

I will cover around 60 services along with their spec for my first milestones

No.	Description of PR	Prereq PR numbers	Target date for PR submission	Target date for PR to be merged
1.1	Make typescript checks strict -I <ul style="list-style-type: none"> ● BaseInteractionValidationService ● CollectionValidationService ● ExplorationDiffService ● ExplorationFeaturesServices (Along with their specs)		8 Jun	11 Jun

1.2	<p>Make typescript checks strict -II</p> <ul style="list-style-type: none"> • GraphUtilsService • ImprovementsService • InteractionSpecsService • MusicPhrasePlayerService <p>(Along with their specs)</p>		10 Jun	13 Jun
1.3	<p>Make typescript checks strict -III</p> <ul style="list-style-type: none"> • StoryContentsObjectFactory • RatingComputationService • SuggestionService • TextInputTokenizer • WindowRed <p>(Along with their specs)</p>		12 Jun	15 Jun
1.4	<p>Make typescript checks strict -IV</p> <ul style="list-style-type: none"> • ReviewTestEngineService • SchemaUndefinedLastElementService • SolutionValidityService • StateGraphLayoutService <p>(Along with their specs)</p>		14 Jun	17 Jun
1.5	<p>Make typescript checks strict -V</p> <ul style="list-style-type: none"> • StateGraphLayoutService • VersionTreeService • ExplorationDiffService • VoiceOverObjectFactory <p>(Along with their specs)</p>	1.3	16 Jun	19 Jun
1.6	<p>Make typescript checks strict -VI</p> <ul style="list-style-type: none"> • UtilsService • VersionTreeService • WinnowingPreprocessingService • AlgebraicExpressionInputValidationService <p>(Along with their specs)</p>		18 Jun	21 Jun
1.7	<p>Make typescript checks strict -VII</p> <ul style="list-style-type: none"> • alertsService • CodeRepValidationService • ContinueValidation • DeviceInfoService <p>(Along with their specs)</p>		20 Jun	23 Jun
1.8	<p>Make typescript checks strict -VIII</p> <ul style="list-style-type: none"> • DocumentAttributeCustomizationService 		22 Jun	25 Jun

	<ul style="list-style-type: none"> • DragAndDropSortInputValidationService • EndExplorationValidationService • ExpressionSyntaxTreeService (Along with their specs) 			
1.9	<p>Make typescript checks strict - IX</p> <ul style="list-style-type: none"> • FractionInputRulesService • FractionInputValidationService • GraphInputRulesService • GraphInputValidationService (Along with their specs) 	1.6, 1.2	24 Jun	27 Jun
1.10	<p>Make typescript checks strict -X</p> <ul style="list-style-type: none"> • GuestCollectionProgressService • HtmlEscaperService • ImageClickInputValidationService • InteractiveMapValidationService (Along with their specs) 		26 Jun	29 Jun
1.11	<p>Make typescript checks strict -XI</p> <ul style="list-style-type: none"> • ItemSelectionInputValidationService • LocalStorageService • LogicProofValidationService • MathEquationInputValidationService (Along with their specs) 		28 Jun	31 Jun
1.12	<p>Make typescript checks strict -XII</p> <ul style="list-style-type: none"> • MessengerService • MetaTagCustomizationService • MultipleChoiceInputValidationService • MusicNotesInputRulesService (Along with their specs) 		30 Jun	1 Jul
1.13	<p>Make typescript checks strict -XIII</p> <ul style="list-style-type: none"> • MusicNotesInputValidationService • NumericInputValidationService • NumericExpressionInputValidationService • MusicNotesInputRulesService (Along with their specs) 		1 Jun	3 Jul
1.14	<p>Make typescript checks strict -XIV</p> <ul style="list-style-type: none"> • NumericInputValidationService • NumberWithUnitsRulesService 		3 Jun	5Jul

	<ul style="list-style-type: none"> • OutcomeObjectFactory • PageTitleService (Along with their specs)			
1.15	Make typescript checks strict -XV <ul style="list-style-type: none"> • PencilCodeEditorValidationService • PlayerTranscriptService • QuestionValidationService • RatioExpressionInputValidationService (Along with their specs)		5 Jun	8Jul
1.16	Adding Documentation for strict types		NA	
	Mid-Evaluation-Period			

Milestone 2

Key Objective:

Introduce typing for 65 twins of files and tests for those files (so 130 files all together).

No.	Description Of PR	Prereq PR numbers	Target date for PR submission	Target date for PR to be merged
2.1	Make typescript checks strict - XVI <ul style="list-style-type: none"> • SetInputValidationService • SkillCreationBackendApiService • SpeechSynthesisChunkerService • SVMPredictionService • CollectionCreationService (Along with their specs)		17 Jul	19 Jul
2.2	Make typescript checks strict -XVII <ul style="list-style-type: none"> • SchemaDefaultValueService • SiteAnalyticsService • StateClassifierMappingService • StateEditorService • LanguageUtilService (Along with their specs)		18 Jul	21 Jul
2.3	Make typescript checks strict -XVIII <ul style="list-style-type: none"> • TextInputValidationService • TopicCreationBackendApiService 		20 Jul	22 Jul

	<ul style="list-style-type: none"> • UrlService • WindowDimensionsService • NumberWithUnitsValidationService (Along with their specs) 			
2.4	<p>Make typescript checks strict -XIX</p> <ul style="list-style-type: none"> • CkEditorCopyContentService • AutogeneratedAudioPlayerService • BottomNavbarStatusService • PreventPageUnloadEventService • PencilCodeEditorRulesService (Along with their specs) 		22 Jul	24 Jul
2.5	<p>Make typescript checks strict -XX</p> <ul style="list-style-type: none"> • CodeReplPredictionService • CodeReplRulesService • ContextService • EditorFirstTimeEventsService • SidebarStatusService (Along with their specs) 	2.3	24 Jul	26 Jul
2.6	<p>Make typescript checks strict -XXI</p> <ul style="list-style-type: none"> • ExplorationImprovementsTaskRegistryService • ExpressionEvaluatorService • ExpressionTagAssembler • FocusManagerService • StateContentService (Along with their specs) 	2.3	26 Jul	28 Jul
2.7	<p>Make typescript checks strict -XXII</p> <ul style="list-style-type: none"> • PencilCodeEditorRulesService • SlidebarStatusService • StateContentService • StateCustomizationArgsService • StateCustomizationArgsService (Along with their specs) 		28 Jul	30 Jul
2.8	<p>Make typescript checks strict -XXIII</p> <ul style="list-style-type: none"> • StateHintService • StateInteractionIdService • StateNextContentIdIndexService • StatePropertyService (Along with their specs) 		30 Jul	1 Aug
2.9	<p>Make typescript checks strict - XXIV</p> <ul style="list-style-type: none"> • StateRecordedVoiceoversService • StateSolicitAnswerDetailsService 		2 Aug	4 Aug

	<ul style="list-style-type: none"> • StateSolutionService • StateWrittenTranslationsService (Along with their specs)			
2.10	Make typescript checks strict -XXV <ul style="list-style-type: none"> • StoryEditorNavigationService • TextInputRulesService • UriInterpolationService • TextInputRulesService (Along with their specs)	1.5, 1.6, 1.8	4 Aug	6 Aug
2.11	Make typescript checks strict -XXVI <ul style="list-style-type: none"> • ValidatorsService • AdminDataService • AssetsBackendApiService • EmailDashboardBackendApiService (Along with their specs)	2.10	6 Aug	8 Aug
2.12	Make typescript checks strict -XXVII <ul style="list-style-type: none"> • ExplorationPermissionsBackendApiService • ExplorationRecommendationsBackendApiService • InteractionRulesRegistryService • AudioTranslationLanguageService (Along with their specs)	2.10	8 Aug	10 Aug
2.13	Make typescript checks strict -XXVIII <ul style="list-style-type: none"> • ConceptCardBackendApiService • CollectionCreationBackendService • CollectionRightsBackendApiService • ContributionOpportunitiesBackendApiService (Along with their specs)		10 Aug	12 Aug
	Final-Evaluations		NA	

Optional Sections

Additional Project-Specific Considerations

Privacy

No, this feature does not collect new user data or change how user data is collected.

Security

No, this feature does not provide any new opportunities for users to gain unauthorized access to user data or otherwise impact other users' experience on the site in a negative way.

Accessibility (if user-facing)

The feature does not impact the UI of the project.

Documentation Changes

- Introduce Documentation for types and common errors faced while making types strict. I would create a separate wiki page for points to keep in mind while strict typing a file. This would also include the sub-types covered i.e strictNullChecks, strictBindCallApply etc. along with FAQs and proper solution of the errors faced while applying strict checks.

Ethics

No, there aren't any ethical considerations related to this feature.

Future Work

- Make typescript checks strict for whole code base
- Introducing Angular CLI Strict Mode since the project would be fully migrated then. Some of the points that this mode will cover are
 - Strict mode in TypeScript, as well as other strictness flags recommended by the TypeScript team. Specifically, strict, forceConsistentCasingInFileNames, noImplicitReturns, noFallthroughCasesInSwitch
 - Turns on strict Angular compiler flags strictTemplates and strictInjectionParameters
 - Reduced bundle size budgets by ~75%
 - Turns on no-any TSLint rule to prevent declarations of type any
 - Marks your application as side-effect free to enable more advanced tree-shaking