Approval Sheet

The Interaction Design Project II entitled “Gamification of Corpus Cleaning”
by Vivek Paul Joseph, Roll Number 156330004 is approved,
in partial fulfillment of the Master in Design Degree in Interaction Design
at IDC School of Design, Indian Institute of Technology Bombay

Internal:

External:

Guide:

Chairperson:

Declaration

I declare that this written document represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/ data/fact/ source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Vivek Paul Joseph
156330004

IDC School of Design,
Indian Institute of Technology Bombay

November 2016

Aknowledgements

I would like to thank Prof. Anirudha Joshi for his support and guidance through the duration of this project. I am grateful to Prof. Girish Dalvi, Prof. Ravi Poovaiah, Prof. Venkatesh Rajamanickam and Prof. Jayesh Pillai at IDC for their valuable feedback and suggestions.

Table of Contents

Abstract

Swarachakra Malayalam is an open source keyboard for android. It generates a steadily growing database of Malayalam words. This word database could be a resource that can be used to fuel development of new tools for the language. But the database(corpus) contains incorrect or unusable words (for certain contexts). Tagging these words becomes an important task to make this corpus usable by 'cleaning' the corpus.

Due to the complexity of the language grammar paired with its agglutinative property, it is a challenging to programmatically categorize the words. But while this may be challenging for a computer, it is easier and even enjoyable for a person who knows the language. But due to the large number of words in the corpus, it becomes a huge task. The aim of this project is to crowdsource, through gamification, the cleaning of the corpus.

During the course of the project, the corpus cleaning activity was broken down into multiple steps and turned in to minitasks. Then multiple possible ways of gamifying these tasks were looked into. After weighing the pros and cons of each, one of the ideas was designed, detailed and developed into functioning prototypes. The prototype version 1 had minimal gamification elements (only level scores and player levels). The prototype version 2 has more gamification elements like scores, player levels, achievements and badges, leaderboards, etc. The proto V1 acts as a benchmark against which player engagement levels of proto V2.

While the proto V2 doesn't have all the gamification elements that were explored, it lays a foundation upon which the others can also be added. The effectiveness of the game in cleaning the corpus and the effect of these gamification elements on player engagement were evaluated using a functioning prototype. Out of the gamification elements that were tried out in the prototype, the tutorial levels, game stats, achievements and leaderboards seemed to have direct positive impact on the players' engagement levels. Identifying the impact of the other gamification elements needs a longer evaluation with a larger user base.

In the bigger picture, the outcome of this project would be one of the several layers of filter that can be used to clean up the existing database and to create a comprehensive database of words.

1. Introduction

Malayalam is the language predominantly spoken in the state of Kerala, India. It is a Dravidian language and is spoken by more than 38 million [8] people. Malayalam has variations based on multiple factors, some of which are region, religion, and community [8]. It also has an abundance of influence, in the form of loan words, from other languages like Sanskrit, Arabic and Hebrew, to name a few. One of the features of Malayalam, which it shares with all the Dravidian languages, is that it's an agglutinative language. This means that every noun or verb can act as a root to which affixes can be 'glued on' to bring out different meanings. Malayalam can have more than 8-10 such affixes, which means that for each root word, there are 1000's of possible variations.[8]. An example of agglutination is given below:

Root Word: ഭാഷ (means *Language*)

Agglutinated Forms: ഭാഷയിൽ (*In the language*)

ഭാഷാവരമുള്ളവർ
(People who can speak multiple languages)

Swarachakra maintains a database of malayalam words that is collected from extracts of texts typed using Swarachakra (Malayalam), a text-input software(keyboard) for android. This word database is essentially created from the words that the users type in using this keyboards. This data can be a resource for development of new tools and resources in Malayalam. Some such applications could be :

- Word prediction feature for keyboards which could potentially help improve text input rates
- Spell Checkers/ Auto correct features: That could reduce the number of errors that users make in text entry.
- Other Word Games: Word Creation from tiles,Crossword games, etc. could be other possible applications that can use the output from the corpus cleaning as its input.

But the data cannot directly be used since it contains errors. Hence, there is a need to identify, tag and correct (if needed) these words (henceforth, these will collectively be referred to as 'cleaning' the corpus). A solution that could do this could potentially be extended to more open source databases to create a comprehensive database of Malayalam words.

Top 10 Incorrect Words			
Word	Frequency	Corrected Word	Frequency of Correct word
എൻറെ (Enre)	707	എന്റെ (Ente)	2877
ഞാന് (Njaanu)	638	ഞാൻ (Njaan)	11565
ചെയ്യണഠ (Cheyyanaddha)	594	ചെയ്യണം (Cheyyanam)	202
ജപിയ്കുന്നു (Japiykunnu)	560	ജയിപ്പിക്കുന്നു (Japikkyunnu)	0
നാഠ (Naddha)	540	നാം (Naam)	1091
യുടെ (Yude)	466	NA	-
നിൻറെ (Ninre)	444	നിന്റെ (Ninte)	1091
എനിക് (Eniku)	354	എനിക്ക് (Enikku)	2333
അന്യാന്യഠ (Anyaanyaddha)	346	അന്യൂന്യം (Anyoonyam)	0
നൽകുകയുഠ (Nalkukayuddha)	282	നൽകുകയും (Nalkukayum)	0

Top 10 Words	
Word	Frequency
ഒരു	13794
ഞാൻ	11565
ഈ	6969
ആ	5713
ക	4756
നീ	4631
അ	4133
അത്	3872
നല്ല	3673
ഹ	3263

Top 10 Unicode Errors	
Word	Frequency
ഫിൻെറ	645
ംം	559
ാാ	513
്	462
ഒാ	440
ോാ	409
ൂ	375
ൊാ	355
അധ്യായഠ	323
ഒാരോാ	308

Fig 1. Overview of Swarachakra Malayalam Corpus

Top 10 Colloquial Usages	
Word	Frequency
ട്ടോ	355
അടിപൊളി	280
ന്റെ	260
എല്ലാരും	250
ചുമ്മാ	249
ഇപ്പം	232
മൊത്തം	150
അപ്പോ	131
ലവൻ	84
പോടേ	44

Top 10 English Words	
Word	Frequency
ഗുഡ് (Good)	678
ബൈ (Bye)	672
ഫോട്ടോ (Photo)	662
ഹലോ (Hello)	556
ലൈൻ (Line)	497
ലീഗ് (League)	377
പോസ്റ്റ് (Post)	358
സർ (Sir)	345
ബ്രോ (Bro)	336
കാർഡ് (Card)	334

It is possible to clean the data using an algorithm that can check these words against the grammatical rules of the language. But the complexity, along with the agglutinative nature of the language makes it very hard to develop such an algorithm; and the huge number of words (in millions) would take a long time for one person or a small group of people to clean it manually. But the same is much easier and even enjoyable if the collective knowledge of the language users is used for the same. Hence, a solution which crowdsources the cleaning of the database can be effective in this case. The database that gets created using such a solution would, then act as a resource for further research and development of tools and resources for Malayalam.

2. Primary Research

The first step was to study the dataset that was available to look for the kind of words (in terms of complexity and context) that are being dealt with here. This is critical, since it would help decide the target audience of the product. The top and bottom (based on frequency) 300 words in the database were examined in this context. It was observed that the words were conversational in nature and moderate in complexity, i.e., a native speaker who knew how to read and write the language would be able to help tag/correct the words. Compared to the words at the bottom of the list, the words at the top were more conversational in nature. The words at the top were, in general, shorter than the ones at the bottom; some of the ones at the bottom having agglutination levels as high as 6. The bottom of the list, as one would expect, had a significantly higher error percentage compared to the top 300.

2.1 The Word Database

As mentioned earlier, word database mostly consists of words that are of conversational nature, which is probably due to the fact that it is generated from the user input to a keyboard (Swarachakra Malayalam). The most frequent words are 2-3 syllable words that are regularly used words. Eg: ഒരു, ഞാൻ, ഈ

The first error shows up at the 148th most frequent word. The error being a substitution error. The database largely consits of agglutinated versions of words and there is the occasional english word (transliterated to malayalam) in the mix.

2.2 Tagging/Correction Activity

The top(based on frequency) 300 words were tagged and experts who have been working on other similar data cleaning projects were consulted. Based on these, the activity was broken down into 3 major stages

- Validate the correctness of the word
- Identifying the type of error
- Correct the words (wherever applicable)

Looking at these steps in more detail, the validation of correctness would involve the player marking a word as correctly or incorreclty spelt. The tagging stage would involve adding further metadata to each of these words.

The tagging itself could be based on various factors:

- Correctness
- Language (Eg: A lot of english words, spelled out in malayalam, were found in the database)
- Root word (This becomes relevant since malayalam is an agglutinative language, thousands of words can stem from a single noun or verb)

- Region of origin (Some words are unique to specific regions of Kerala)
- Other grammar related metadata : These would prove helpful in identifying grammar rules.

Examples of word tagging/correction:

Word	Category	Type of Error	Corrected Word	Origin
ഒരു	Correct	-	-	-
ഹലോ	English	-	-	-
ജപിയ്കുന്നു	Incorrect	Substitution	ജപിക്കുന്നു	-
എൻെറ	Incorrect	Unicode	എന്റെ	-
ലവൻ	Slang	-	-	Trivandrum

Since the followup data connected to a word depends on its category, the categorization tagging becomes an important step in the process.

To find out the kind of tags people were leaning towards, 4 native language speakers from IDC were asked to tag the top 50 and bottom 50 words in the database. The categories tha emerged from this exercise were: ‘Correct Malayalam’, ‘Incorrect Malayalam’, ‘English Word’, ‘Incorrectly Spelled English Word’, ‘Colloquial Usage’

Since the number of words was limited, the list of categories generated may not be all encompassing. But for the initial iterations of the the database cleaning, these categories were chosen.

No. of Users	4
No. of Words	100
No. of Unique User Tags	7
User tags (Standardised)	**Count**
Correct	232
Incorrect	88
Slang/Regional Dialect/Colloquial Usage	32
English Word	27
Incorrectly Spelt English Word	8
Not Sure	13
Total Cumulative for 4 users)	400

Fig 2. Summary of user responses for the word tagging activity

3. Secondary Research

In the secondary research stage, the main focus was on examining the existing word datasets and tools related to Malayalam that can be used to aid the project; research papers on existing crowdsourcing projects utilizing gamification elements; and the study of casual games from PlayStore to try and identify features that work for them.

3.1 Existing Malayalam Word Databases/Tools

There are three major word datasets available online for Malayalam: The 'Datuk Corpus'[7] (Fig 3), Malayalam WordNet [13](പദശൃംഖല) and Shabdkosh:Malayalam[14](Fig.4). While the Datuk corpusand shabdkosh are crowdsourced in public domain, the WordNet was crowdsourced to a controlled group (It is currently not online either). But being dictionary datasets, they contain the most of the root words and a few random agglutinated forms of the words. Hence, their utility in filtering out the words in the Swarachakra dataset iimited, since the words there are mostly in agglutinated form.

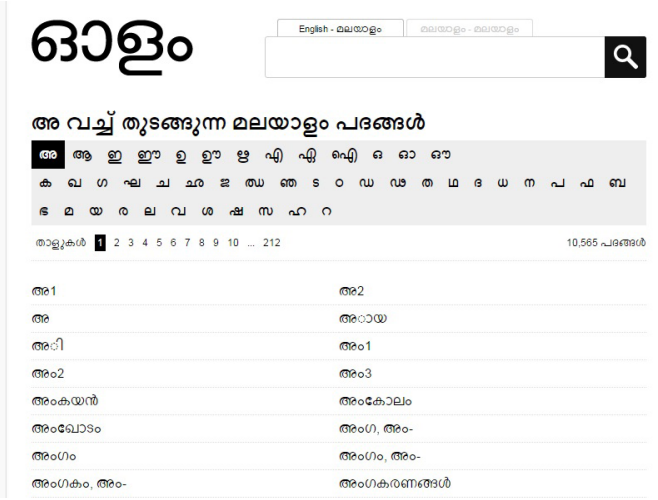


Fig 3. Olam Corpus

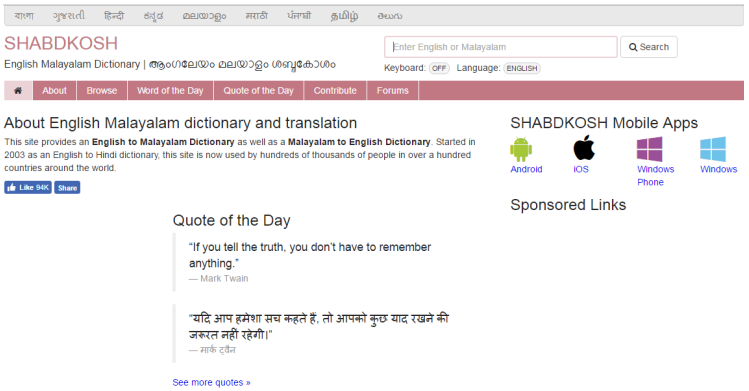


Fig 4. Shabdkosh

Other tools like Malayalam Spell-checkers (Eg: https://goo.gl/pkpkND) are available online which could potentially be used for filtering incorrect words. But they tended to fail when the agglutinated forms of words are fed in. (Fig 5: The word that is entered does not exist, but the system doesn't mark it as an error) Out of 100 words(from the corpus) that were fed into this system, it identified 4(out of 23) errors correctly and wrongly marked 12 more as incorrect.

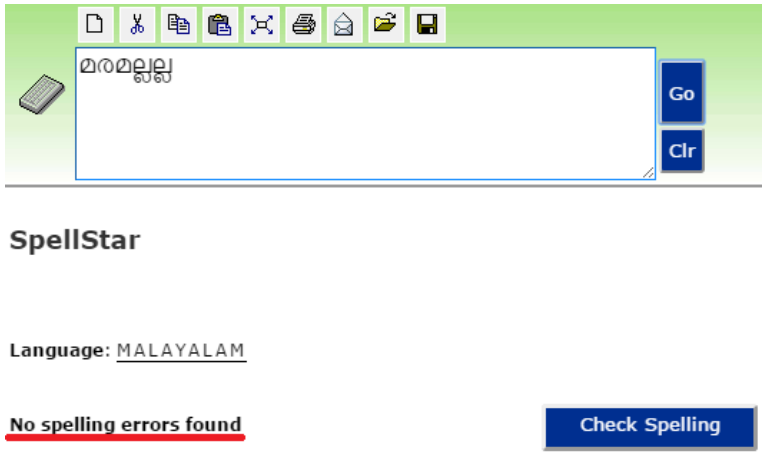


Fig 5. Online Spellchecker (SpellStar)

3.2 Gamification

Gamification is the application of game-design elements to an activity to improve the engagement of the user. It has been in use in various fields like science, health, education, image/audio tagging, etc. [1][3] Gamification ideally involves adding gameful elements to an activity, enhancing them through motivational affordances of that are similar to that of games. [1] Some of the motivational affordances under gamification are: points, leaderboards, achievements/badges, levels, story/theme, feedback, rewards and progress[4]

Gamified experiences have been used for data collection/validation/ tagging. Some notable examples are ESP Game, Waze and Peekaboom. ESP Game(Fig 6) is an image tagging game which aims to add as many tags as possible to an image. It is an online game where two players are shown an image and both try to guess what tag the other has given to an image. When both of them type in the same string, it is called a match and they move to the next image. It was seen that player engagement increases when the activity is competitive/collaborative in nature. Bringing in these elements into the corpus cleaning activity could potentially increase the participation of the users and thereby yeild great effectiveness for the product. ESP game [3] which was used for tagging of images available in the internet. The only gamification element used here is multiplayer play and scores, which proved to be sufficient to motivate players to contribute, some even spending over 50 hours. A key takeaway is that competitive play could be a major motivator in crowdsourced projects.

The task at hand in this project can also be approached from similar angles. The steps of the corpus cleaning can essentially be translated

Fig 6 Image Source: Google Image Search

into a series of tagging tasks. Since the gamification elements of multiplayer matches and scores have proved to work in such tasks, the same can be attempted here. On top of these, more gamification elements can also be incorporated in order to enhance the player engagement further.

An interesting gamification model was the freemium model [4], where the player does task based activities to earn points which they then use as a tokens or in-game boosts while a game. The main game generally has nothing to do with the research task as such. This model has been tested and proved to work for crowdsourcing of laboratory tasks.[4] Though, it must be noted that there was a dip in quality of crowd response i(n the gamified version compared to the lab test) when the tasks were cognitive in nature.[4] The main takeaway here is that the tasks that are presented to the user (as part of the corpus cleaning) should be low on cognitive requirement.

3.3 Study of gamification elements in existing casual mobile games

The approach here was to study the existing top rated/played games in the android Play Store; and identify the features that people like in them. These features could then be incorporated to the solution being developed. Top games from different categories in the Play Store were studied. The categories looked at were Top 16 language/ word related games and Top 11 of all the games. The games were selected based on top games' lists in websites and PlayStore's own rankings. These games were studied in terms of complexity and features.

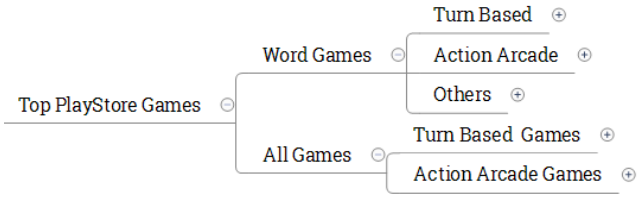


Fig 7. Game classification followed for study

A few of these games and their features are discussed here.

Word Game: Alphabear

Premise: Player creates words from given letters (tiles on the board) to score points. (Fig 8)

Key Gamification elements: Points, unlockable levels, multiplayer, tournaments, achievements, unlockable collectibles(score boosters : introduces an element of strategy as well. Player can choose three boosters at the beginning of the level)

Design Implication: Elements outside of the main task can be used to give the user extrinsic motivation to play the main task

Fig 8. Alphabear
Image Source: Google Image Search

Word Game: Wordiest

Player is given 14 letters from which they are to create two words. Their submissions are compared (Fig 9) against other players' who were given the same set, which adds an interesting multiplayer element to it, without having direct player-player contact.

Design Implication: The users can effectively be put in a multiplayer setting without them having to play at the same time. This could help increase the number of ways multiplayer modes can be incorporated into the design.

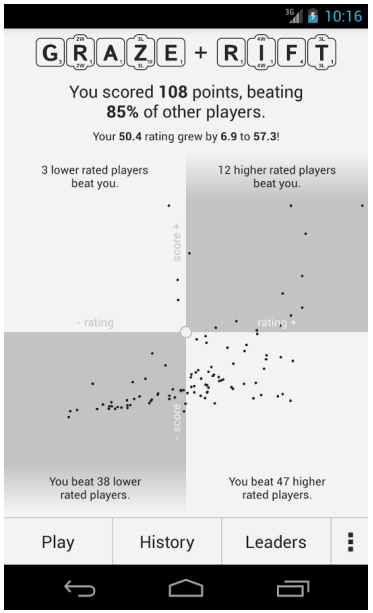


Fig 9. Wordiest- Visualization of player performance
Image Source: Google Image Search

General Game : Crossy road

The game involves players avoiding obstacles on a procedurally generated course. With this simple premise, the game uses external gamification elements like collectibles (bought with in-game/real currency), highscores and leaderboards to engage players.

Design Implication: If the basic mechanics of the play are simple to understand, users would still play the game (even if repetitive) in order to meet external goals (like collectibles, highscores, etc.). Also, providing a (real world) currency price tag to collectibles adds a value to them, thereby generating further motivation amongst users to unlock them (there needs to be a way that the user can unlock them by putting in play hours instead).

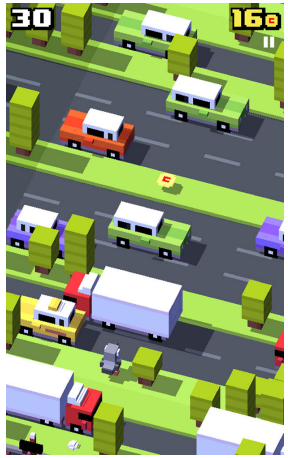


Fig 10. Crossy Road
Image Source: Google Image Search

General Game : Candy Crush Saga

Players match 3 tiles of a kind on a board (Fig 11) in order to score points and complete objectives (specific to level). The game has 100s of unique levels, with a clear path of progression (Fig 12). There are also collectibles that can be earned by scoring more points

Design Implication: Such a level structure would provide the player a sense of progression as they play different levels (even though the gameplay largely remains the same in most levels).

Fig 11. Candy Crush Saga - Gameplay
Image Source: Google Image Search

Fig 12. Candy Crush Saga - Level Progression
Image Source: Google Image Search

Key observations from this study can be summarized as:

- One notable observation was that the games that were purely word based (like the ones mentioned earlier) have a considerably less number of downloads than other games, even if they both offer similar gamification elements. On an average, for all the games that were checked, the ratio in number of downloads was almost 10:1. This suggested that general games possibly has a larger audience than purely word based games.
- Features like collectibles, social media integration, unlockable levels/modes, player-vs-player modes, leaderboards and resource collection are found in some form or the other in all the top games.
- The more complex a game gets, the less number of players play it. For example, games like crossy road, candy crush, etc, which have simple mechanics and very little strategizing required seem to have a higher number of players than their more complex counterparts. Here, by complex, I mean games where the player needs to choose between powerups or level modifiers to maximise score/clear levels.
- Arcade games are more likely to get downloaded and played than purely word games.

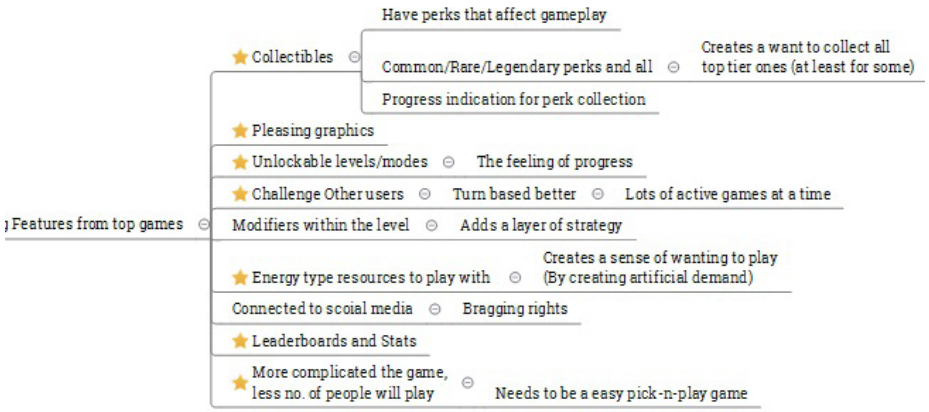


Fig 13. Excerpt from study of game elements

4. Ideation

Based on the observations and findings from the previous stages, an initial set of ideas was generated. The attempt in the beginnning was to keep the gameplay arcade instead of being purely word/language based. Some such ideas (discussed in this section) were not detailed out since they would have been too big a task to be implemented and tested out within the duration of the project. The selection of ideas for detailing was partly based on its potential to engage the user and partly on the feasibility to be detailed, developed and tested out during the available time (duration of the project). Four of these ideas that were looked at in some depth are presented in this section.

4.1 Puzzle Game

Initial set of ideas were based on freemium model (as discussed in section 3.3 of this report), where the main game would be an arcade puzzle game. The player would need to collect resources for completing the main game by completing secondary tasks (dataset cleaning) (Fig 14). One of the ideas explored under this model is discussed below:

The main game is a puzzle, which requires the player to traverse on the screen from one point to the other. The player needs to use red and blue crystals to use the paths of the corresponding colour. The element of strategy on the player's part is to choose the shortest path by which they can reach the end. And then gather the crystals required for the same and clear the level with those. The crystals are gathered by cleaning the corpus : red for tagging and blue for correcting words in corresponding tasks.

The player can choose to avoid strategizing and brute force their way through the level, if they have sufficient gems. Hence, the player gets to choose their playstyle.

Pros:

- Can reach out to a larger range of audience
- Gives freedom in terms of game concept for the primary game, which would be independent of the database cleaning task (secondary game)
- The levels can be procedurally generated to ensure that the player gets a unique level in every round.

Cons:

- Too large a scope to be completed within the duration of the project

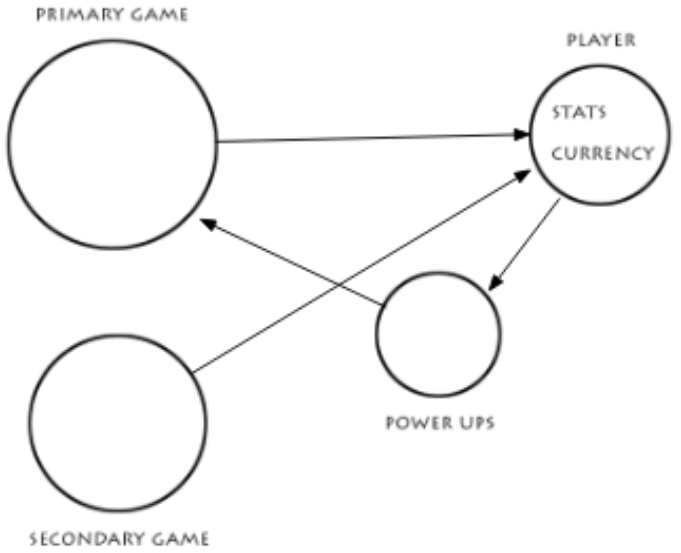
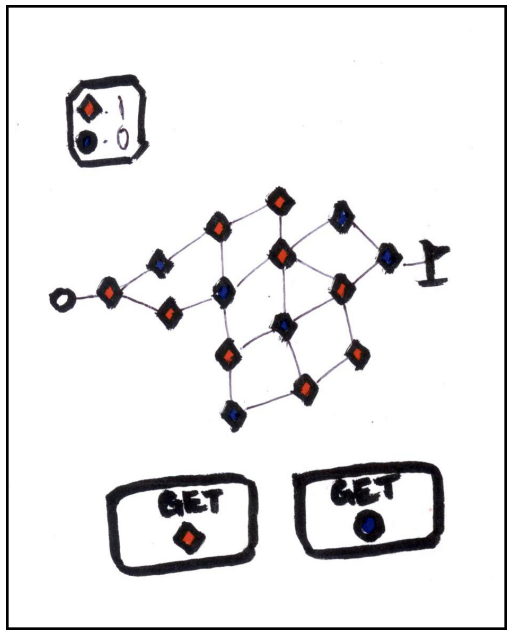
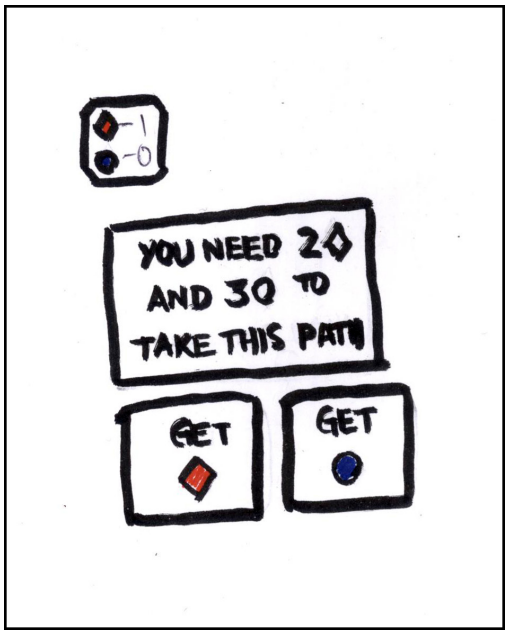


Fig 14. Game Overall Structure

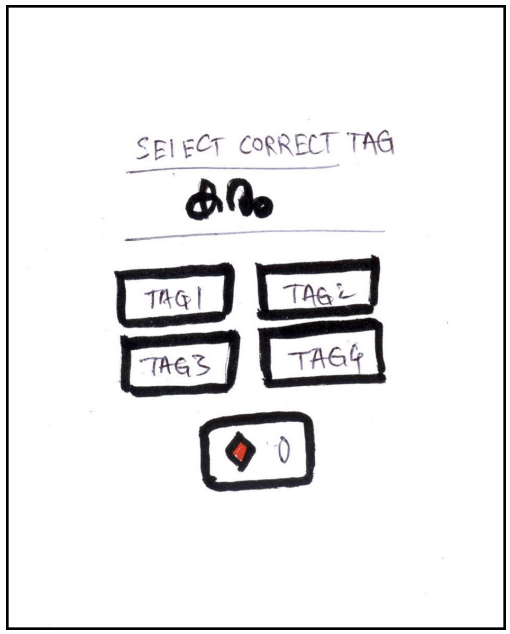
Gameplay Scenario



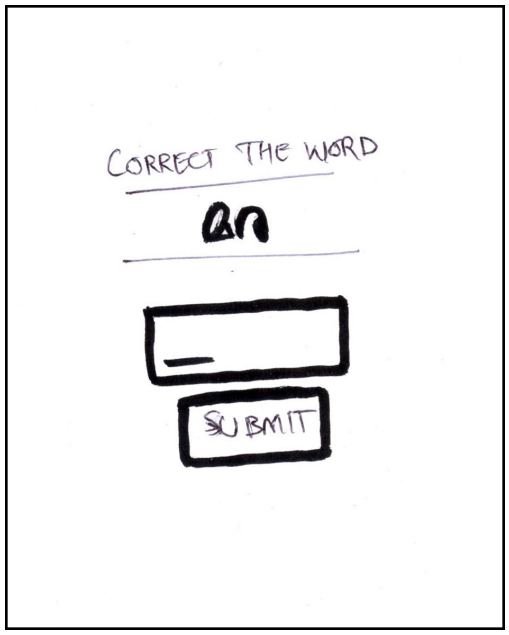
Player Examines the puzzle and identifies the shortest path possible



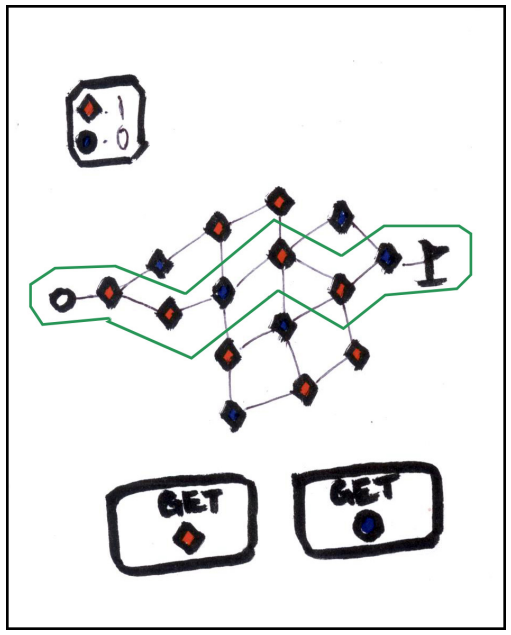
Player selects the path and the game tells him that he needs 3 more red crystals and 2 more green ones



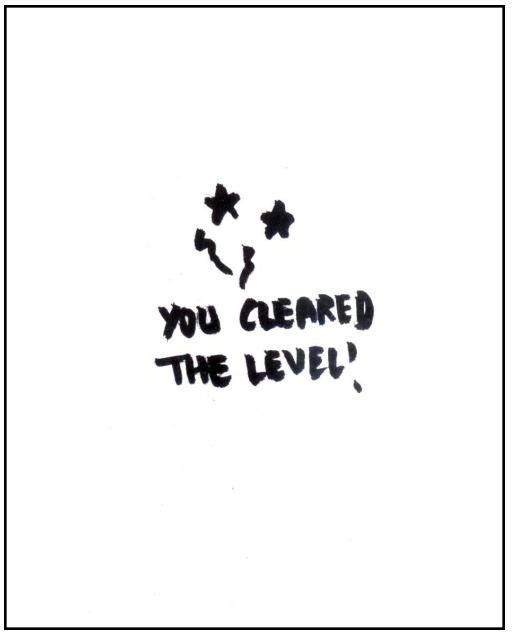
Goes to tagging task. Tags enough words to collect the required red crystals.



Goes to Correcting Task screen and corrects enough words to collect required blue crystals



Heads back to the puzzle board and submits the crystals to open up the path



Wins the round and moves on to the next

4.2 Role playing Game

A loot based game where players goes around a map collecting scrolls(words). These words can then be sold or fixed to build keys, which can then be used to open chests that are scattered around the map. Opening these chests would give them stat boosting items.

The players battle each other with spells, which are cast using ‘intelligence’ points. Hence a player with higher int points would be more likely to win than one with lower.

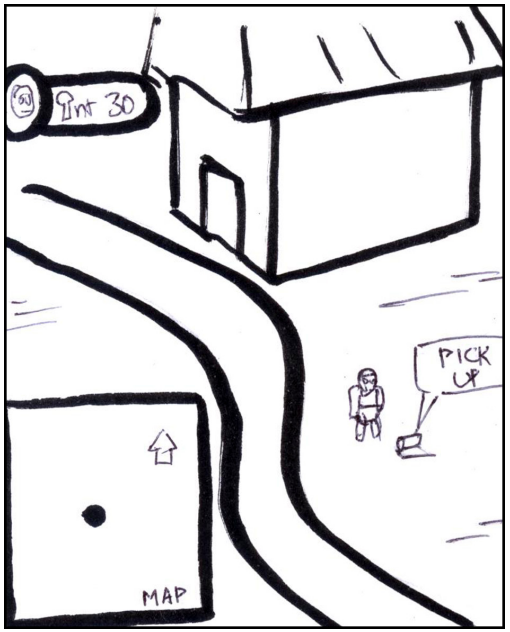
Pros:

- Since the gains from the word tagging/correction has direct effect on the player performance, it would act as a good incentive.

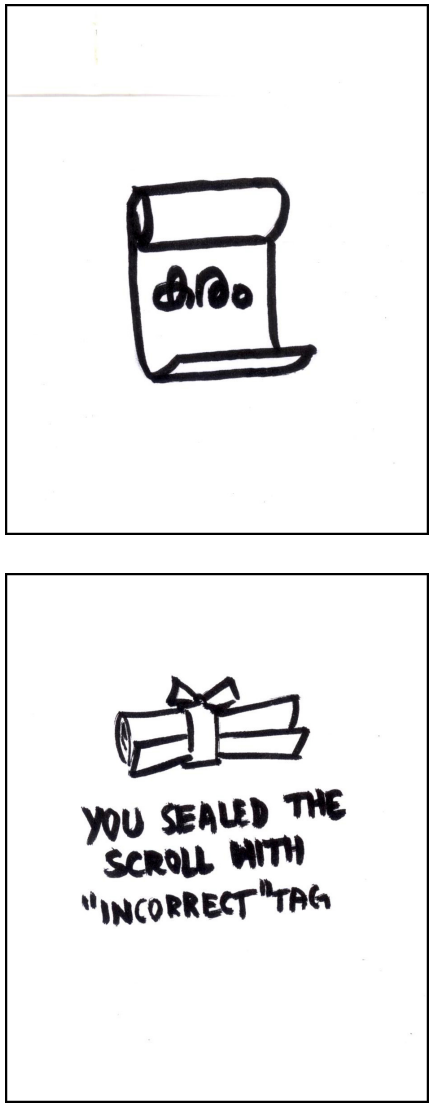
Cons:

- The rate of words getting corrected would presumably be low/ moderate, since there are multiple other activities that need to be done in the course of the game
- Scope may be too large to get to a evaluable stage within the time constraints

Gameplay Scenario



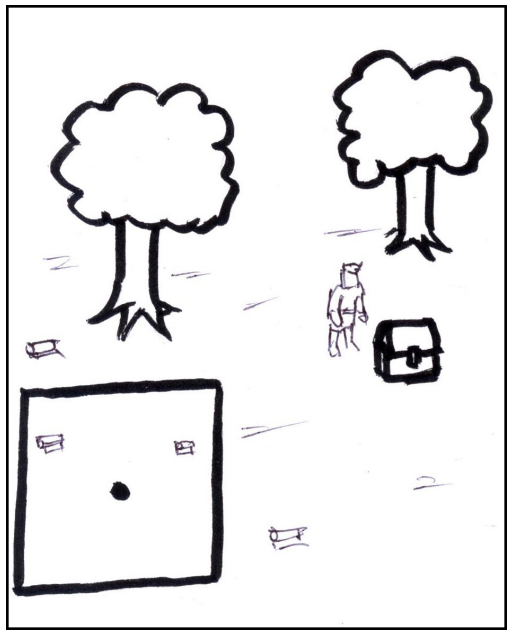
Player walks around map to find a scroll



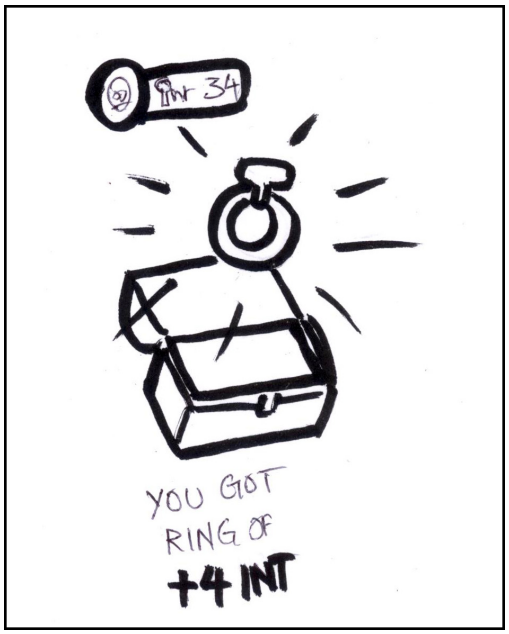
Player opens a scroll and finds a word. If the word is not correct, the player tags it with the appropriate tag and then proceeds to sell it.

He sees that the word is a correct one so he goes to a store to sell it for gold,.

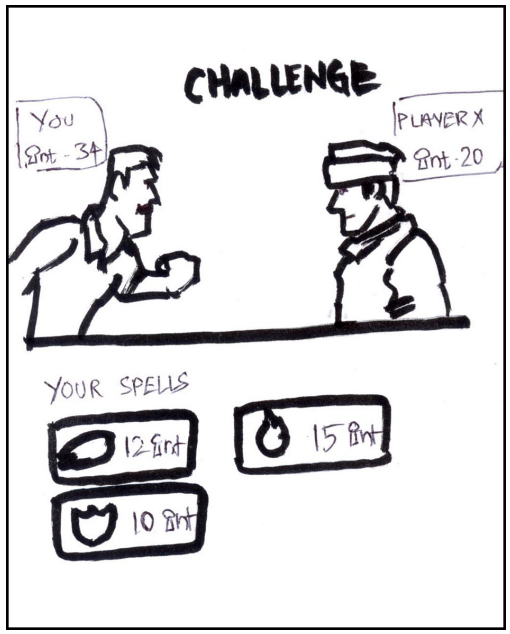
He uses the gold to buy a key



Player walks around the map to find a chest



He uses the key to open the chest and gets an item that grants him +4 int points

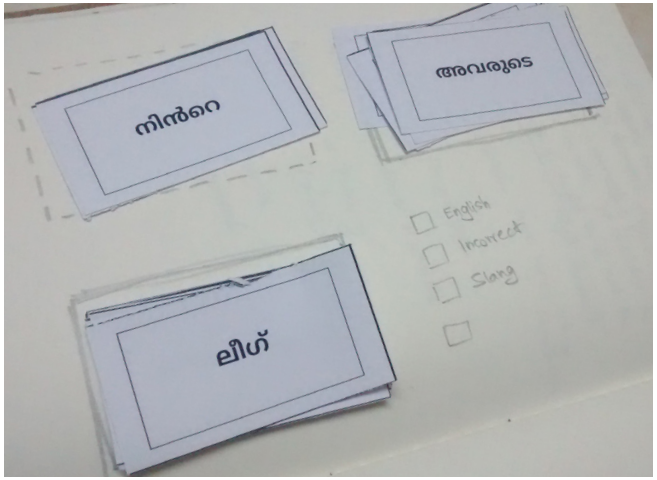


He challenges another player for a duel and since he has higher int points, he can cast more spells and wins

4.3 Word Trade

The players are word traders who buy words from travelling salesmen (computer controlled) and sell the words at in-game marketplaces, in exchange for currency. The player would make profit out of correct words and loss out of incorrect ones. This currency is then used to buy more words and so on. Tagging words and fixing incorrect words at the in-game ‘garage‘ would fetch them more currency.

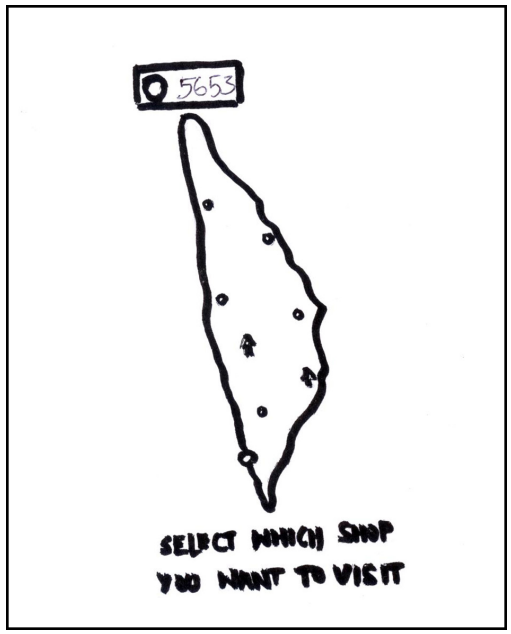
This idea was tested out using a paper prototype and tested with 4 users (further tests were scrapped based on the feedback from the first 4 users)



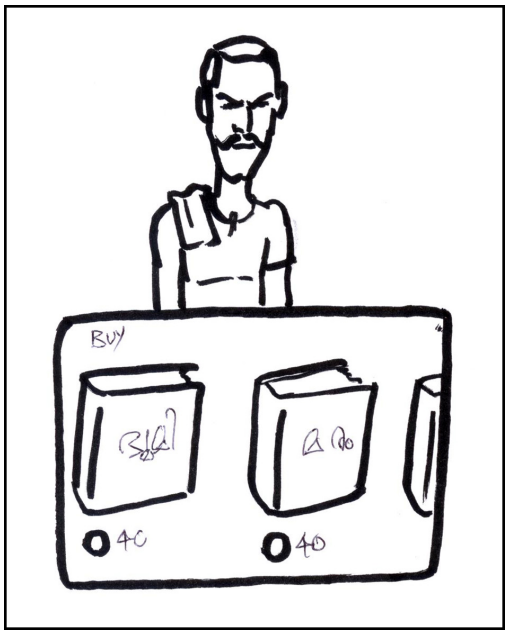
Observations & Player Feedback:

- Buying/Selling was an unnecessarily lengthy process. The player is likely to get tired of it soon (it was visible during the test stage itself, which was a short period already)
- Needs more challenges. The words in the dataset are too easy
- There is no intrinsic motivator. The player felt disconnected from the knowledge base building, which is one of the key motivators for the entire activity.

Gameplay Scenario

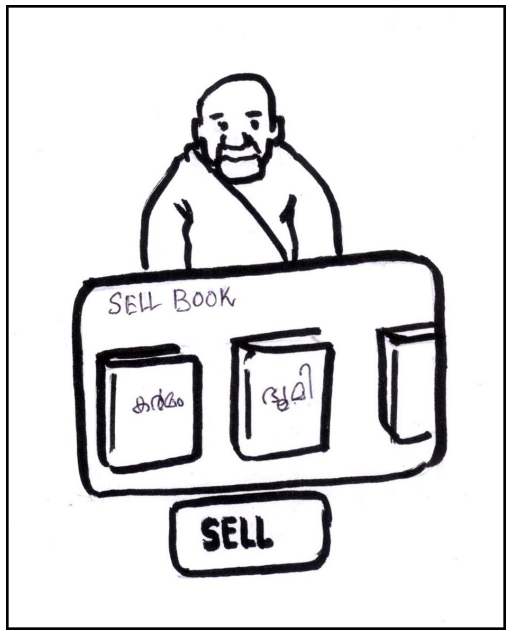


Player looks at the map and looks at which shop to visit



She visits a shady travelling salesman. These characters sell books at a low price. But the book could turn out to be a fake (i.e., the word could be incorrect).

She buys two books from him



She goes to a trader who buys and sells books at the regular price to sell off the books she bought earlier

She tries to sell one of the books. Its a genuine one! So she makes a profit.

She sells the other book. But it turned out to be a fake and she made a loss.

4.4 Word Master

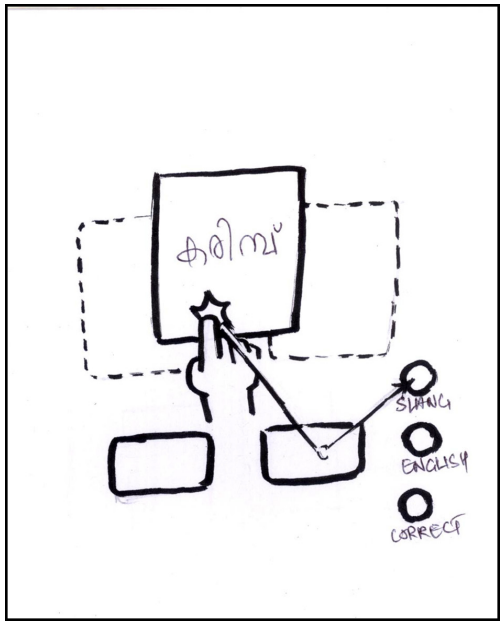
The feedback from the previous idea was incorporated to form this one. Essentially a task based game, the approach is quite direct here. The players would tag/correct words in exchange for points. They will be rewarded for correct answers. The different activities involved, i.e., tagging and correction would be done through different game modes.

Pros :

- The players are briefed from the start that they are contributing to the knowledgebase, which would act as a good motivator to players who take pride in their language. The task would be inherently enjoyable to these players
- The task based approach would ensure a faster rate of tagging/ correction compared to the other modes

Cons

- Since it is largely task based, other elements need to be added to make it fun/challenging to play
- Based on the pros and cons of each idea and keeping in mind the implementation, WordMaster was chosen to be detailed out



5. Final Concept

After weighing the pros and cons of the different ideas, WordMaster : Malayalam was chosen and detailed out. Some aspects that were focused on during the detailing of the concept were:

- Try and keep reminding the player that they are contributing to the knowledge
- To have a variety of modes that the player can choose from so that the user doesn't get bored of playing the same one over and over
- Avoid delayed gratification as much as possible. This becomes a challenge here, since the game would not be able to evaluate if the player's response is correct till it has sufficient data.
- Have a user data evaluation system that can adjust to different kinds of user behaviour so as to generate reliable output.

5.1 Game Modes

As mentioned ealier, the word database cleaning activity was split into muliple steps. These steps were then distributed into different modes of gameplay that would be available to the player to choose from. Each mode serves a specific function in the word data cleaning process and the output from one mode is used as input to another.

5.1.1 Proving Grounds Mode

Function : Player proficiency measurement

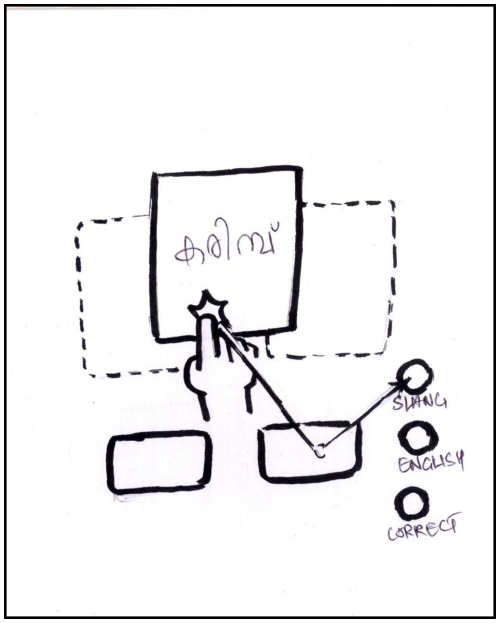
The player is asked to categorize the words into predefined categories. They are rewarded points for correct responses.

The player is shown the words one by one and presented with a set of buttons that correspond to each category. Player presses the button that they think is the correct category. Immediate feedback is given to the player, where they can see if they got the previous word correct or not.

The categories chosen (based on the categorization activity discussed in primary research section) were: ‘Correct’, ‘Incorrect’, ‘English Word’, ‘Incorrectly spelled English Word’ and ‘Slang’

The word dataset used in this mode is already tagged and hence the correct category is known to the system. The players’ responses are compared to the correct tag to evaluate the correctness of the player response.

The dataset contains 150 words which are distributed over 15 rounds, each round having 10 words each. The distribution into rounds brings a break in the activity and gives the player feedback on their performance. It is also meant to give the player a sense of progression.

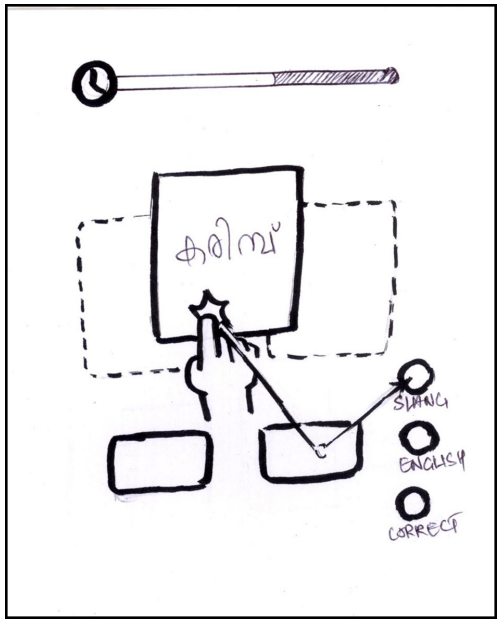


5.1.2 Basic Single Player Mode

Function: Word tagging/validation

This mode is similar to proving grounds. The player is presented with one word at a time, to which they respond by choosing one of the presented categories. But here there is a time restriction imposed on the player. When the time ends, the round is finished and at the end of each round players are shown a review screen which acts as immediate feedback to their performance.

This round acts as the first step of tagging of the realtime data. Unlike proving grounds, which uses pre-tagged words as input, this mode uses uncategorized data. Hence, when the player inputs their response, the system wouldn't know whether the player response is correct or not (till sufficient number of users have responded... correct response estimation discussed later on in the report). This presents the challenge in the form of delayed gratification, which could be a potential turn off for the players (As noted during the paper prototyping of earlier ideas).



5.1.3 Challenge Mode (Multiplayer)

Function : Word Tagging/Validation

The player plays against a human opponent online in a game of tagging words. There are added points for being quick. After each player attempts 10 words, their scores are compared and a winner is announced.

This mode tries to tap into the competitive nature of the player, which has proven to be an excellent motivator in gamified crowdsourcing projects[1]

5.1.4 Error ID-ing Mode

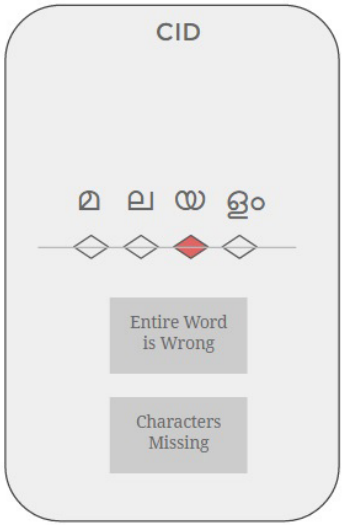
Function: Word Correction

Word Source : Basic Single Player Mode

The words that are marked as “Incorrect” in the basic modes are fed into this mode, where the player is shown the word chunked into groups. The player, then, needs to identify where the error lies and mark it/them.

The aim of this mode is to make the word correction process more enjoyable than directly asking them to type in the correct word. The players who are more inclined to just correcting the word rather than playing this mode are given the option of marking the whole word (instead of identifying the exact location of error).

The responses from multiple users will be pooled and the most likely error location is then sent to the next mode.



5.1.5 Error Fixing Mode

Function: Word Correction

Word Source : Error ID-ing Mode

The output from Error ID-ing mode would be fed into this mode, where words would be shown to the player with the erroneous part (as identified by players in the previous mode) blanked out and the player is asked to fill in the blanks with what they think is appropriate. The player types in their response.

The mode would cater to the different possible kinds of text input errors, i.e., substitution, insertion, omission and combinations of these. The output from this mode would be a corrected word against each input word.

The responses recorded from all the players would then be pooled and a most likely response would be evaluated. This will then be used as the ‘potential’ corrected form of the word (that was tagged incorrect to begin with)

‘5.1.6 This or That’ Mode

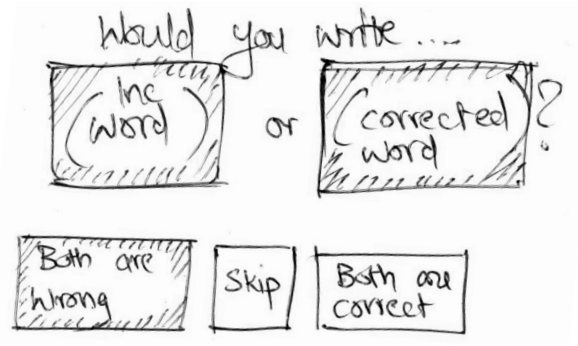
Function: Validation

Word Source: Error Fixing Mode

The last step of the word tagging/correction process in the game. The incorrect and ‘corrected’ versions of each word are presented to the player and they are asked to choose which one is correct.

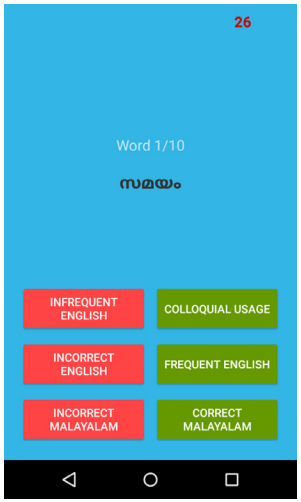
This mode would be more effective for validation than the basic mode since the player would be able to compare the correct and incorrect versions to make their decision.

If the players choose (system estimated most likely response from the pooled player responses) the corrected version that was generated using the Error-Correction mode, then the tagging process is marked as complete and the wod is fed into the out put database.



5.2 Concept Prototype V1

The first prototype presented the player with the ‘proving grounds’ mode. The data was already tagged and the users were given exp points for getting their responses right. A levelling system utilized the exp points to increase the player level. Scoring and levelling systems were the only gamification elements added in this prototype. The goal, here, was to test out the core game mechanics.



This proto would also set a benchmark for user engagement, which can be used to compare against second proto, which would have more gamification elements.

5.3 Data Design

Data design here refers to the design of the storage, flow and interpretation of the data. It is critical to the game generating the desired output dataset. Figure 16 shows the flow of data between the different game modes.

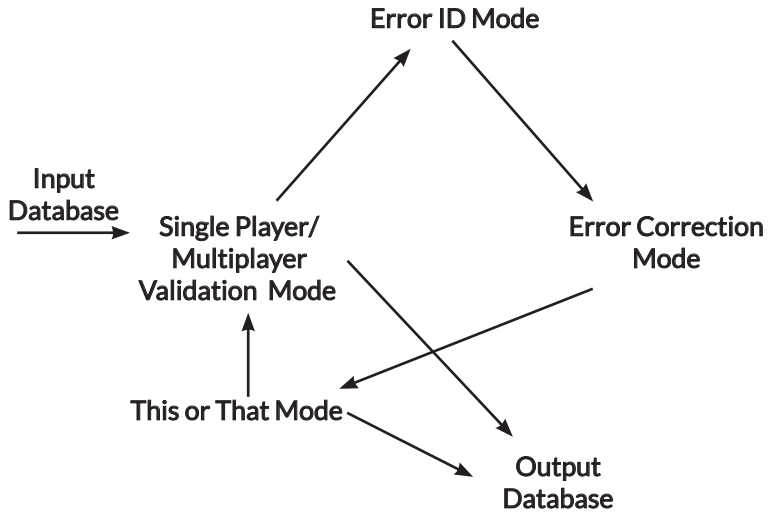


Fig 16. Data Flow between modes

Evaluation of Crowdsourced Data

Crowdsourcing uses the inputs from the users in building knowledge (in this case, cleaning of the database). The correct answer is evaluated by either averaging user responses or using the majority response as correct (applicable in this context). But the final answer could get biased because of incorrect responses from some of the members of the crowd. Such responses may be because of them not understanding the task, being lazy or malicious. Hence member quality estimation becomes critical for the success of the crowdsourcing system. The crowd response evaluation system should be designed in a way that it gives less value to low quality responses and higher value to high quality responses. Also, at some point, the low quality members could be given feedback in an attempt to help them improve their responses in the future.

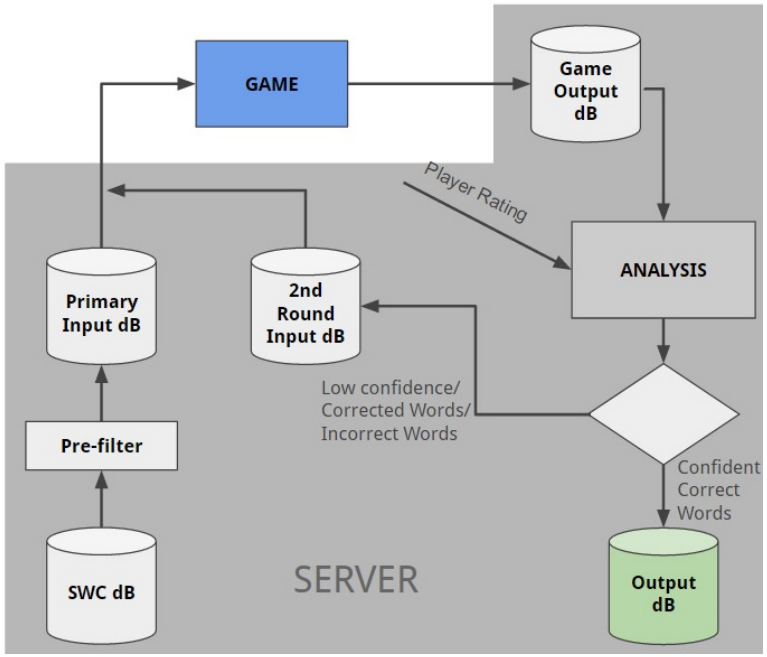


Fig 17. Data Flow between device-server

Another challenge that presented here is that unless there are sufficient responses from the crowd, the game wouldn't know what the correct response should be. This is more prominent at the launch of the game when the number of users is also less. While the reward for such words can be given later, the delayed gratification would have a negative impact on the player motivation. So in such cases, half of the points are awarded initially and the rest after it has been categorized by the system.

As for the correct response evaluation algorithm, the logic followed is : once there are sufficient number of responses for a word, the category which has >90% of the responses that players have marked would be adjudged the correct response. The algorithm was developed keeping in mind that the players' proficiency should be taken into account at any given point of time. A more proficient player's response would carry more value than that of a less proficient one. The game determines the player proficiency in the gold standard levels and then continuously monitors it.

The data that gets generated by the game would be finallly stored in the below format:

Word	Validity	Validity Confidence	Category	Secondary Category (Optional)	Type of Error	Corrected Word	Correction Method
XXXX	[Correct, Incorrect]	[0-1]	[Malayalam, English, Slang]	[English, Slang]	[Unicode, Substitution, Other]	XXXX	[Manual, Crowdsourced, Automated]

Table 1: Final Schema for data storage (specific to malayalam)

This format can be used for the final corpus that is generated by the game (or from other sources). Further details of the data stored by the game or any other source of corpus cleaning can be stored separately. The words (corrected word) that go above a threshold on the v'alidity confidence' (Ref Table 1) can be, then used for generating reliable corpuses for the language.

5.4 Prototype V2

This prototype had two modes : proving grounds and single player. While the proving grounds round remained more or less similar to the V1 prototype, the single player mode worked with uncategorized data. The gamification elements included in this prototype are:

- Points & Levels
- Player Statistics
- Feedback on performance
- Online Leaderboards & Ranking
- League System for multiplayer
- Achievements and badges

Contribution Points (CP) & Level System: The choice of words is to remind the user at every point that they are contributing to the knowledge base. The levelling system which is driven by CP. Care had to be taken to ensure that the CP requirements for any level is neither too high, nor too low. This is critical, since if it is too high, levelling becomes too easy and loses value. If it is too low, then the user would get frustrated.

Player Statistics: Different facets of the player performance would be tracked and made accessible to the player so that the players who are inclined towards it can identify and improve their numbers. (Fig 18)

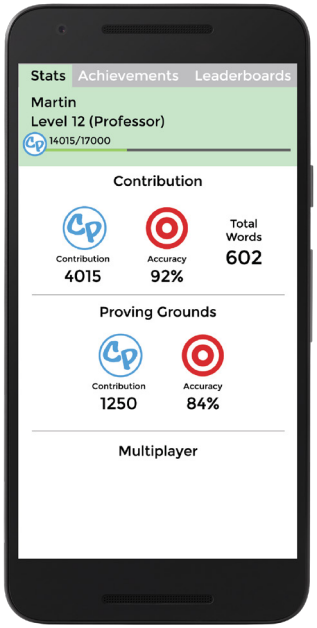


Fig.18 Stats Screen

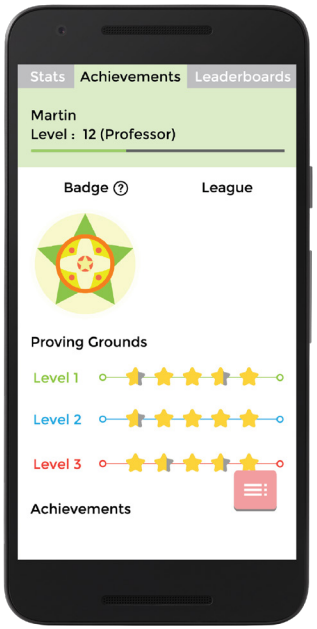


Fig .19 Achievements Screen

Feedback: Once there are sufficient statistics available on a player's performance, they will be given feedback in the form of messages that try nudge them to improve in the particular area that they are currently lacking in.

Achievements: Achievements would act as small goals within the game, completing which would provide the player with a sense of prorgression. (Fig 19)

Online Leaderboards & Ranking : In the prototype the leaderboard tracks CP and ranks players accordingly. Once the game has more modes, other stats like multiplayer scores, wins and rankings can also be used (Fig 20)

Badges: Each player gets a badge that gets personalised based on their stats and achievements. This badge is then displayed in multiplayer modes and leaderboards. (Fig 21)

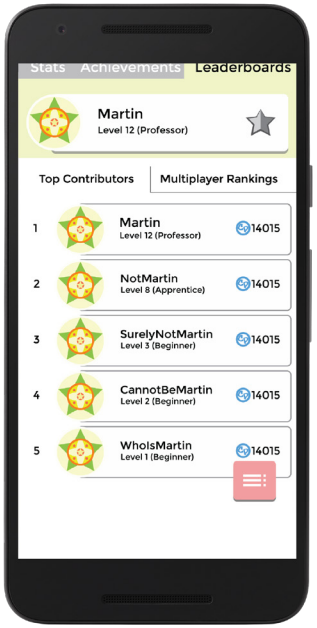


Fig. 21 Leaderboard Screen

Data Segregation

Since the game tracks a lot of player data and word data, it becomes important to choose which ones to present to the player and the mode of representation. A hierarchy of data was created as a thumb rule to the data presentation. An attempt was made to keep minimal information at surface level. More detailed information is made available in dedicated stats screen. As much information as possible was conveyed through images over text in an attempt to reduce the cognitive load on the player.:

Tier One Information (Presented on every screen with the userdata)

- Name
- Level
- Tag Based on Level
- Badge
- PvP* Pass Count
- PvP* League

Tier Two Information (Highest Priority Info on Specific Stat Page)

- Contribution Points
- Accuracy
- PvP
- PvP Exp
- League
- Ranking (leaderboard)
- Achievements
- Level Based - Indirectly Contribution based
- Global Ranking Based
- Percentile based

*PvP: Player vs Player (Multiplayer mode). The player has to earn passes to play the PvP mode by playing the single player mode.

- Rank number based
- Accuracy Based
- PvP Exp Based
- Attempted Words Based
- Proving Ground Performance
- Global Stats
- Leaderboard Ranking
- Contribution
- PvP Ranking
- LeaderBoards
- Top Contributors
- Top PvP Ranking

Tier Three Information
(Data with Lower Priority in Specific Stat Screen)

- Words attempted
- Words correct
- Rate of contribution
- Global Stats
- Contribution (percentile)
- PvP
- No.of matches
- No. of wins
- Win Rate

UI Elements

The tier 1 information were presented to the player at almost every screen since they are the key data pertaining to the player's profile. The data's representation was made graphical in nature over textual. This was for two main reasons:

1. To reduce the cognitive load on the player
2. In case of specific data like the contibution points and levelling, the graphical representation gave a better sense of progress compared to textual

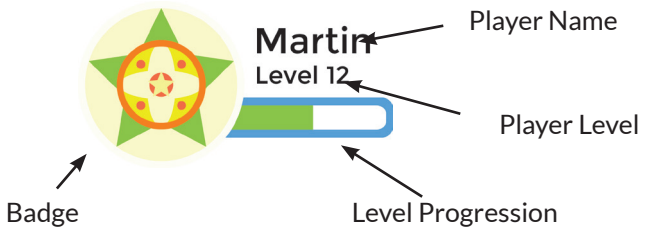


Fig 22. Tier 1 Data Widget

Badges: The badges are used here as a visual indicator of the player's accomplishments. The badge is used as the player's avatar in the leaderboards (Fig 23) and multiplayer modes. The badge is personalised based on the player's play style and performance.

This is done by splitting the badge into layers and each layer being set based on a different parameter. Fig X shows a 3 level badge system.

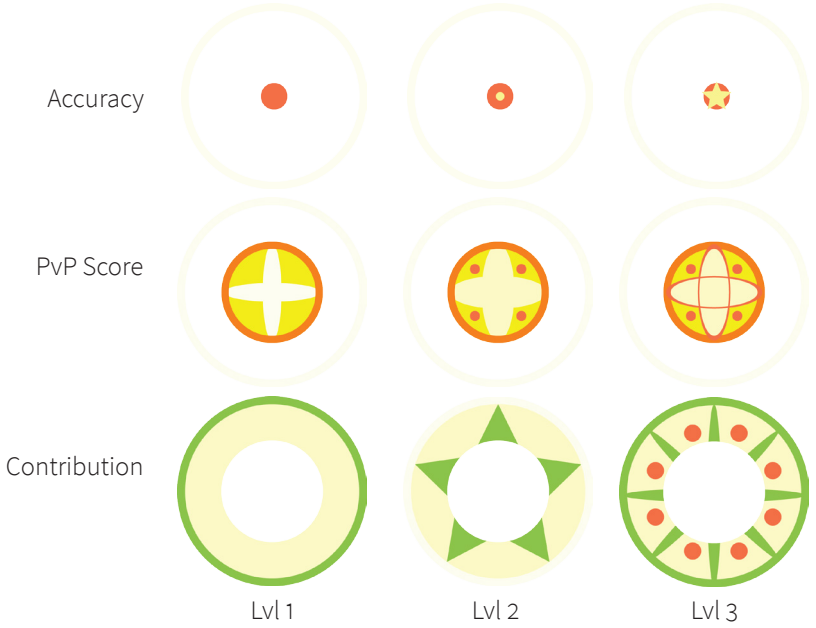


Fig. 23 Badge System

Every player's badge is a permutation of the options. For example, if a player has earned:

Accuracy: Level 3

PvP Score: Level 2

Contribution: Level 1

Then the corresponding badge would be:

Level Timers (In-game): The level timers were made visual in order to free up the cognitive capacity to correctly and quickly tagging the word.

Wireframing and Flow Design

The different individual UI elements were integrated into the different screens in the flow. The data to be displayed in each screen was as planned during the data design stage.

All the frequently used buttons were placed in the bottom 2/3rds of the screen. This is to cater to players with phones that have >5" screen size since it becomes hard for them to reach these buttons. Most of the recently launched phones come under this category. The top 1/3 of the screen was used only for displaying information.

The final screen layouts were first done in Adobe Illustrator and the elements were then imported into Android Studio, which was used for the game development.

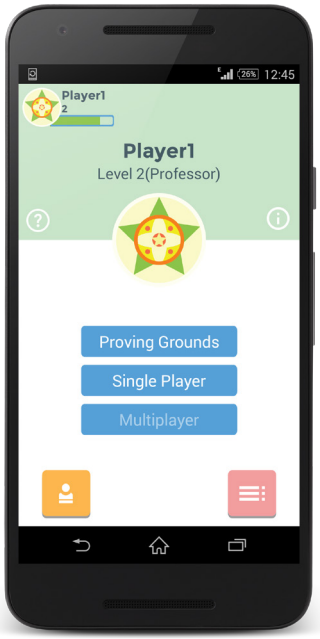
Development/Coding

The development was done from scratch using Android Studio/ Java. An online server was set up using firebase. This is a realtime database that feeds untagged words to the player's device and collects their responses.

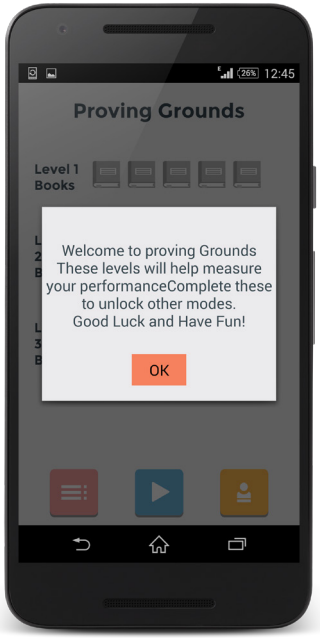
In order to avoid data corruption due to multiple players trying to update it at the same time, a software (maintained by the admin) would be used to periodically update the stats of a word.

The prototype has two (of the six) modes, which are detailed out and are functional.

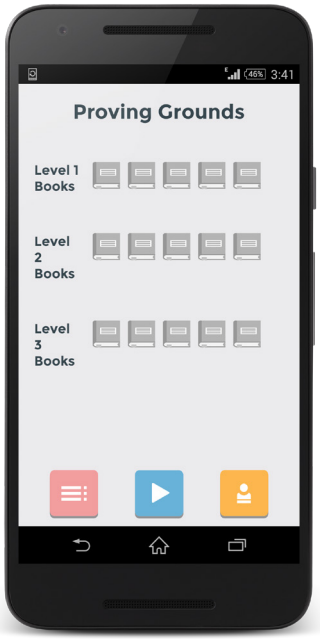
5.5 Gameplay Scenario (For the Prototype)



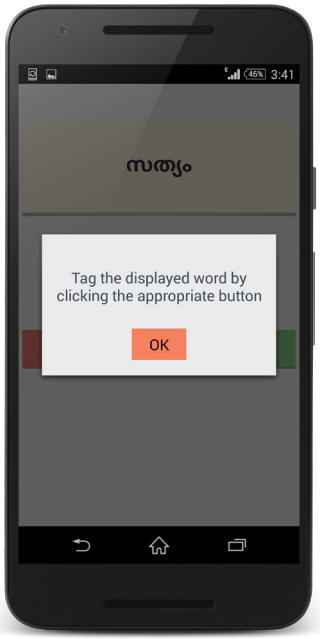
Player launches the game. He can see his stats and level on the main menu



He clicks on ‘Proving Grounds’ since other modes are locked right now. The game tells him that he has to complete these levels to unlock other modes

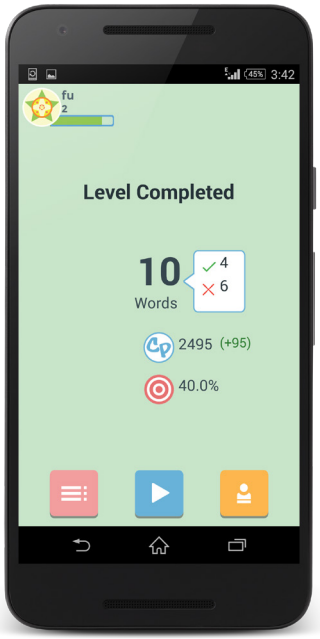


He sees the list of levels and presses the play button to launch the game.

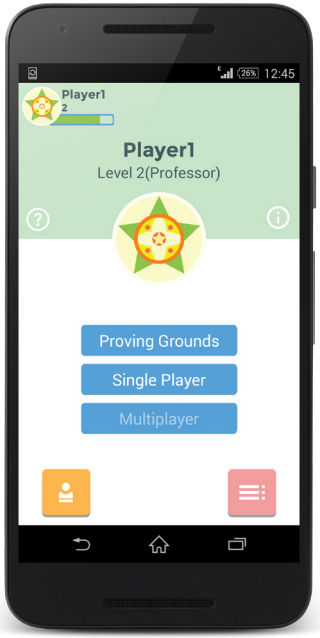


Before the level starts, he is shown an interactive tutorial which tells him what to do in the course of the level

Player plays the tutoiral levels, which progressively increases the number of categories from which he can choose one to tag the word with (from 2 to 5).

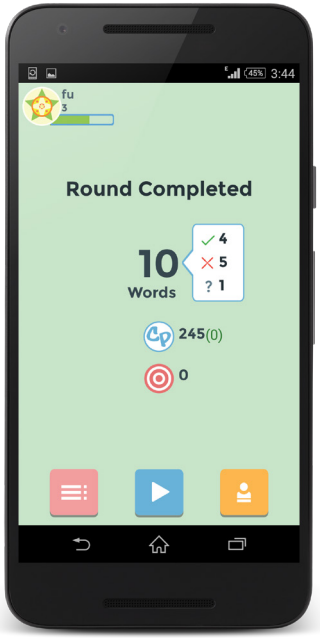


At the end of the level, he is given feedback for his performance in the round and his overall performance.

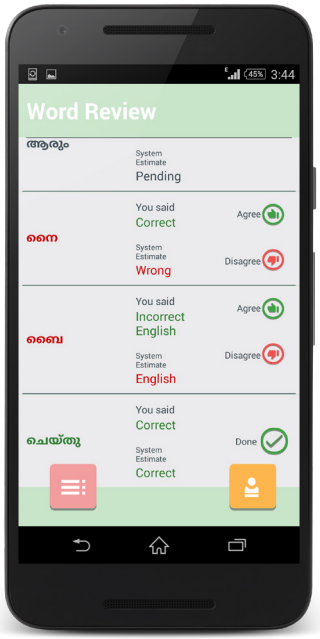


After completing the proving grounds, he unlocks single player and selects it from main menu

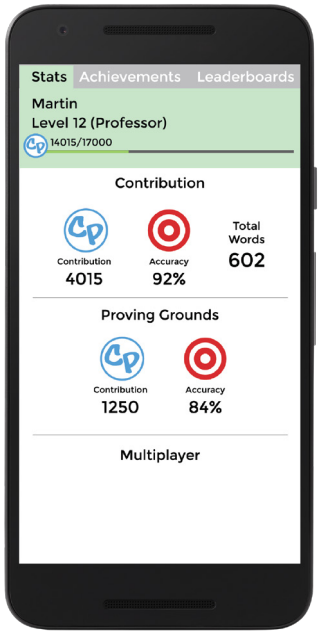
In the single player game, he has a timer to race against and get as many words as he can. But he has to be careful to not make mistakes as his score and stats would drop, then.



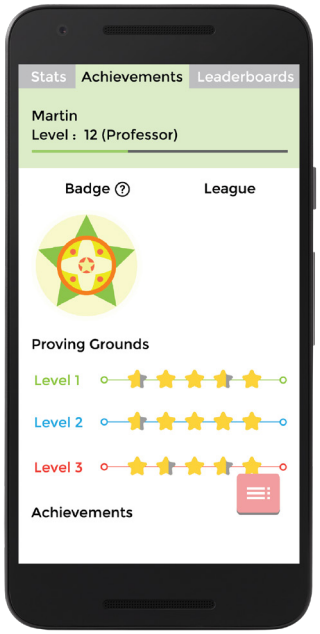
He reviews his performance in the round



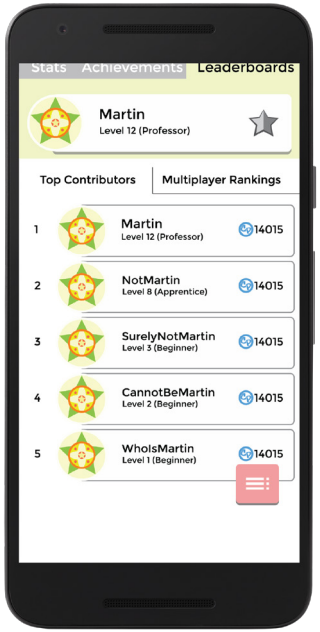
He goes to the word review screen where he sees which words he got right and which he didn't. For the ones he didn't, he gets to upvote or downvote the system estimate.



Now the player goes to the user stats screen where he can get more details on his performance



He swipes right to go to review his achievements in the game



Now he swipes right again to see the leadeboard where he can compare his score and badge to other players and see where he stands in ranking

6. Evaluation

The prototype needed to be evaluated for its effectiveness in cleaning the dataset and its engagement levels. The effectiveness was evaluated using the error rate in categorization that the game makes; and the rate at which the game categorizes.

The evaluation plan consisted of testing the prototype with at least 20 players. First, the user would attempt the Proving Grounds mode, where their proficiency was measured. Apart from the 150 words in the 'Proving Grounds' mode, the players were provided a set of at 500+ words that was tracked over a period of 3 days. As the users played the game, the rest of the framework used their inputs to categorize the words.

The game was evaluated mainly on 3 fronts:

- Ease of Use
- User Engagement
- System Effectiveness (How accurately the overall system could classify words)

6.1 Usability

he aim of this step was to check if the users are able to understand what to do and to identify what issues they were facing. In this stage, the app was installed in 5 users' smartphones and they played the game from scratch, in the evaluator's presence. TThe users' feedback was taken through a mix of think-out-loud and observation.

The key points that came out of this step were:

- Initial tutorial levels: These levels are meant to introduce the player to the kind of words/errors that they might come across. The player can't cross these levels without giving the correct response
  - Novice (at the language) players found them helpful to understand the goal of the game and what kind of errors to look for. They made errors, i.e., marked incorrect ones as correct, in their first run. (Eg.: Similar looking characters like 'ం' and 'ం' often confused the player).
  - Expert players didn't seem to need them. They could identify the errors right away.

- Proving Grounds and Single Player Mode
  - Occasionally, players clicked a category different from the one they wanted to pick. Most such cases were when a word of a different category follows a series of words of the same category. The timer also seemed to be another factor adding to the frequency of these instances, since more such cases were observed in the timed modes.
  - To avoid this, provision needs to be given for retrospective correction between two consecutive response.
- Single Player Mode
  - All players gave the feedback that a new level started abruptly, i.e., right after they press play, the level starts. A countdown timer (3...2...1) would help alleviate this issue.
  - 2 Players (out of the 5) did not see the timer running during the level. The timer needs to have more attention grabbing features (Eg. Blinking towards the final 10 seconds).
- Word Review Screen
  - Collecting points for each word one-by-one was too tedious to the users. A “Collect All” button would help.

6.2 User Engagement

In this stage, the game was distributed to 25 users who would play the game at their own leisure. They were not informed about the game being an evaluation so that it doesn’t effect the amount of effort they put in. The users used the app over the course of 3 days. The aim of this step was to check how much time the users spent on the game over this period.
The key observations are as given below:

Number of Active Users :19 (out of 25)
Test Duration : 3 days

Total Daily Engagement: 3.5 hours
Daily Engagement per user: 19 min
Sessions per user: 2.4
Average Session Duration: 8 min

Overall Number of Responses (Weighted) : 11,166
Average Weighted User Response Rate : 245 per day per user

Note: At the starting stages, the rate at which the words get verified (based on the crowd input data) would be low, since the number of users is low. Hence, at a lot of points, there were too many words that were pending verification from the system side. This was one of the factors that discouraged the users to play more. This should ideally get better as time progresses and more users join.

On the flip side, the engagement values obtained were possibly affected by the novelty factor of the game. It needs to be evaluated over a longer period (ideally with more game mode) for a better estimate.

6.3 System Efficiency

This step is an extension of the previous one. Here the performance of the categorization (done by the overall system) was evaluated. Over the course of the 3 days that the game was being handled, the user inputs were collected and used to categorize the words. The categorized words, were, then cross-verified with an expert.
Key observations are given below:

Number of Active Users :19 (out of 25)

Test Duration : 3 days

Number of words Categorized by the system: 159

Number of words Correctly Categorized: 159

System Accuracy : 100 %

Rate of Categorization : 2.79 per day per user

(This roughly translates into around 280 words being categorized per day with a pool of 100 players.)

The number of unique words in the Swarachakra Malayalam Corpus (at the time of the conception of the project) was **326216 words**. With a player pool of **1000 players**, the game would be able to categorize all of the words in the corpus in **116 days.**
Note: The algorithm that feeds the words to the users was not programmed to take words out of the system as and when one

got confidently categorized by the system. This resulted in a huge number of responses for the initial set of words but much less for the later words. Adding this features would significantly boost the rate of categorization further.

Moreover, the evaluation was conducted on a prototype of the game, which didn’t have all the game modes or the gamification elements that were planned in the overall design. Especially, the multiplayer mode is likely to boost the average amount of time spent per person, thereby boosting the rate of corpus cleaning.

7. Future Scope

The prototype that was created in the course of this project implements one game mode and a few gamification elements. The game, thus does the validation and tagging part of the data cleaning process. For it to be able to do the word correction, more variety in game modes (as mentioned in the ‘Game Modes’ section (5.2) of the report) is needed. This would also improve the user engagement.

Once the game is developed and proven for Malayalam, it can be extended to other languages where similar requirements exist. These language variants would all exist in a common ecosystem of badges and top scores. This would add a layer of competition between players of different languages, potentially increasng the user engagement.

Another future development could be to add modes that would help crowdsource the identification of grammar rules for the language. These rules could then be used to programmatically clean the dataset.

8. Conclusion

During the course of the project, a framework was designed and developed, which crowdsourced the corpus cleaning activity by breaking it down into multiple gamified tasks. Two prototypes were developed. These prototypes were in the form of an android based game that was installed in the users' phones. The prototype version 1 had minimal gamification elements (only level scores and player levels). The prototype version 2 had more gamification elements like scores, player levels, achievements and badges, leaderboards, etc. The prototype V2 was also linked to an online server, so that the crowd data could be analysed in real time and update the corpus. The prototype V2 was evaluated on usability, user engagement and corpus cleaning performance fronts. 3 days of evaluation data demonstrated that the system could reliably categorize the words at a rate of 2.79 words per day per user.

Bibliography

[1] K. Huotari, and J. Hamari, “Defining gamification: a service marketing perspective”, In Proceedings of the 16th International Academic MindTrek Conference, October 3-5, 2012, Tampere, Finland, ACM, pp. 17-22.

[2] J. Hamari, J. Koivisto and H Sarsa, “Does Gamification Work?”, In Proceedings of the 47th Hawaii International Conference on System Science, 2014

[3] L. Ahn and L. Dabbish, “Labeling Images with a Computer Game”, In Proceedings of CHI ‘04: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, April 24–29, 2004

[4] K. Dergousoff and R. Mandryk, “Mobile Gamification for Experiment Data Collection: Leveraging the Freemium Model ”, In CHI ‘15: Proceedings of Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, April 2015

[5] M. Joglekar, H. Garcia-Molina and A. Parameswaran, “Evaluating the Crowd with Confidence”, In CHI EA ‘12: CHI ‘12 Extended Abstracts on Human Factors in Computing Systems, Austin, Texas, USA — May 05 - 10, 2012

[6] W. Lasecki and J. Bigham, “Self-correcting Crowds”, In KDD ‘13 Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, Chicago, Illinois, USA — August 11 - 14, 2013

[7] “The Datuk Corpus — **ദതുക് മലയാളം പദാവലി.**” The Datuk Corpus — **ദതുക് മലയാളം പദാവലി**. Web. 12 Nov. 2016.

[8] “Malayalam.” Wikipedia. Wikimedia Foundation. Web. 12 Nov. 2016.

N. Hung & D. Thang,, “Minimizing Efforts in Validating Crowd Answers”, In Proceedings of SIGMOD’15, May 31–June 4, 2015, Melbourne, Australia.

[9] J. Chamberlain, “The Annotation-Validation (AV) Model: Rewarding Contribution Using Retrospective Agreement”, In Proceedings of the First International Workshop on Gamification for Information Retrieval, Amsterdam, The Netherlands — April 13 - 13, 2014

[10] L. Galli, P. Fraternali, A. Bozzon, “On the application of Game Mechanics in Information Retrieval”, In Proceedings of the First International Workshop on Gamification for Information Retrieval, April 2014

[11] H. Roinestad, J. Burgoon, B. Markines, F. Menczer, “Incentives for social annotation”, In Proceedings of the 20th ACM conference on Hypertext and hypermedia, June 2009

[12] D Johnson & J. Gardner; Personality, motivation and video games”, In Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction, November 2010

[13] Malayalam WordNet. (2016, May 02). Retrieved Sept 19, 2016, from https://en.wikipedia.org/wiki/Malayalam_WordNet

[14] SHABDKOSH| **ശബ്ദകോശം** (2016, July 12). Retrieved Sept 19, 2016, from http://www.shabdkosh.com/ml/