SayIt
A Speech Assistant for the Differently Abled

Manikanta Reddy D.[1]
Supervised by Prof. T. V. Prabhakar[1]
[1]*Indian Institute of Technology, Kanpur*
manikant@cse.iitk.ac.in, tvp@cse.iitk.ac.in

Keywords: Ionic Framework, Cerebral Palsy, Cross Platform Application, Reversed Configuration

Abstract: Speech and Communication disorders are common in people with cerebral palsy. Due to restricted movement in oral-facial muscles, oral-articulation becomes difficult. It becomes important to have an alternative mode of language. Handwritten notes, with pre-decided quotes, play the role of passive communicators, initiating little communication.
We target this problem by building a cross-platform mobile application to assist the specially abled, our user. The application is built using Ionic Framework and provides for two different modes of assistance with Speech capabilities. The application is built to help the people, who assist the user, in building their own personal notes for the user and sharing them.

1 Introduction

Cerebral Palsy is known to affect individuals by constricting the muscles, thereby restrict their movement. It becomes hard for them to move the facial muscles to speak. Hence it is very common to associate palsy with speech and communication disorders.

In most the cases the affected is assisted by others. The assistant usually try to communicate by pre written notes to be displayed as answers for questions. The verbal mode of communication is thus transferred into sign based conversation. This is a great mode of conversation and is widely used. But this method is limited to the amount of physical cards one can carry.

We aim to target this problem by building a mobile application that can emulate the card based mode of communication.

2 Application Details

The application aims to provide a way for assistants to design their own personal *actions*. In this way the assistant and affected are able to communicate in a jargon of their understanding.

By shifting the weight of design towards the users themselves, we make the application more personal, and unique to the individual's requirements.

3 Technology

The application is built on top of the *Ionic Framework*. The following features of the framework make it desirable to wrok with.

- Free & Open Source
- Fully Cross-Platform
- Built on top of
  - Apache Cordova
  - Angular 4
- Write once, compile for all

Ionic Framework uses TypeScript, that compiles to javascript and generates HTML views that are natively binded to the mobile OS, elements by cordova.

Typescript is superset of javascript aiming to provide for better type checking compatibility and error correction during compilation. It minimizes runtime pitfalls that are fairly common due to javascript's ability to accept any type.

Typescript also makes it easy to integrate in ide, thus enabling us to write better code, that's extensible.

4 Design

SayIt was built keeping the personal bond between assistants and the disabled in mind. Hence it is necessary to make the application be usable by both of them. The key features of the design are,

- Mobile Focused
- Support for Gestures
- Disjoint modes of building and using
- Facility to share
- Reversed use case design

The disjoint modes enable the assistant to build the actions required in a normal fashion, whereas the disabled can use the app in a specially built mode, that enables them to select things with large margins of error. It makes it easy for them operate the application.

4.1 Data Model

The data model of SayIt is built to be hierarchical. This choice was made as it is easy to cluster a certain actions into a group called category.

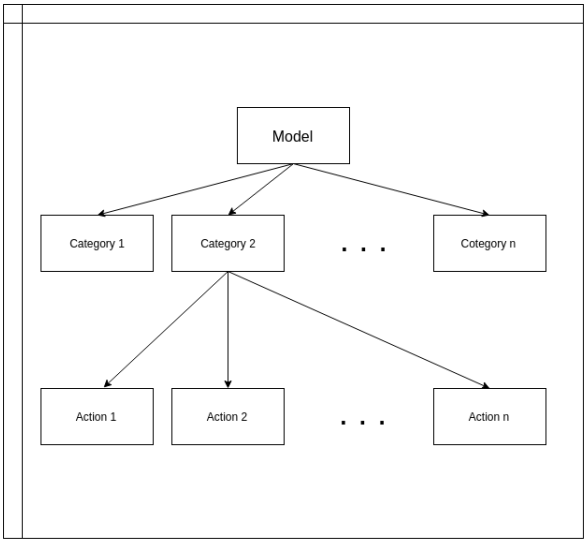

Figure 1: Hierarchical model of data

Every category node has an list of actions it groups together. The categories as well as the actions are built by assistants. The build mode of the application facilitates to edit the nodes as well.

The data model is stored in the *internal storage* construct of HTML5 browsers. The persistence is limited to the scope of the application and is purged when the application is un-installed.

Every node also consists of an entry called image, which is the path to image of the action. The images

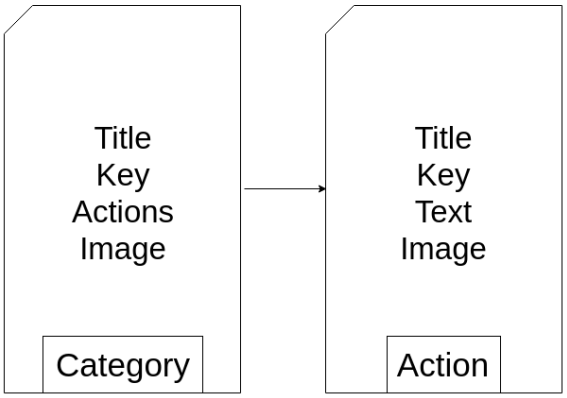

Figure 2: Hierarchical model of data

can be taken either by camera or picked up from library. The images are stored in the local storage as well.

4.2 Engine

SayIt is driven by a data tree traversal engine. The engine is primarily responsible for performing Breadth First Search on the data tree to retrieve information. The engine takes advantage of Angular's data binding abilities to update models to dynamically change the views.

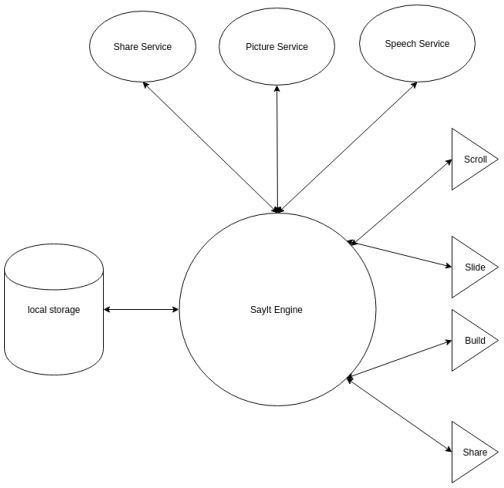

Figure 3: SayIt Engine

The Engine supports two modes of execution.

- Builder
- Viewer
  - Scroll Mode
  - Slide Mode

4.3 Sharing

The engine also supports sharing of the actions. A user can simply select the actions they want to share and send them via social media or bluetooth or other sharing options available on their device.

Sharing is managed by a share service, that packages the actions by stripping their images and preparing them for easy installation on the target device. The actions are then hashed and it is sent along with the data file.

When a valid data file is loaded it is first checked if the data matches the hash it is packaged with. This ensures no junk data is packed into the app. If the hash matches, this data file is certainly generated by the same application on a different device.

4.4 Application Status and Testing

The application is currently under testing by single individual, affected by cerebral palsy, whose feedback for the design has been invaluable. The application is open-sourced and is available over github. Repository

5 Conclusion

As an undergraduate project we've built an application targeting the problems faced by individuals suffering from cerebral palsy. SayIt provides a intuitive interface for the specially abled to express themselves, with the help of the people that assist them, with less dependency. With ready to speak actions the individuals can say what they have in their mind with less difficulty. By being a cross platform application, we have tried to provide the service to as many needy as possible.

Acknowledgements

We thank Dibya Ranjan Sahoo for his invalubale support in building the application by testing and providing feedback.

We thank Prof. T. V. Prabhakar, for having this idea in my mind for the good of the disabled and providing us with an opportunity to build the application.