

Reading Trainer Bobby

LAUREA MAGISTRALE 1st Year

ADVANCED USER INTERFACES

Jalil Esseffar



POLITECNICO
MILANO 1863

Abstract

The primary goal of the "Reading Trainer Bobby" project is to create an innovative reading trainer designed to assist children with reading difficulties. This is achieved through the generation of personalized social stories using OpenAI's technology, which aims to enhance reading skills and comprehension. The system generates custom social stories tailored to the individual needs and reading levels of children, helping to engage them more effectively. The trainer can adapt the complexity of the stories based on the child's progress, providing a scalable learning experience that grows with the child. This project is crucial for addressing the needs of children with reading difficulties by providing an interactive and adaptive tool that makes learning to read more accessible and engaging. By leveraging OpenAI's advanced language models, "Reading Trainer Bobby" offers a novel approach to improving literacy, potentially transforming reading education for young learners.

***"Practice the philosophy of continuous improvement.
Get a little bit better every single day." – Brian Tracy.***

Meet the team:

Jalil Esseffar	jalillesseffar@gmail.com +34642234352 ERASMUS	
----------------	---	---

Member Contribution:

Jalil Esseffar	Development of web application Backend/Frontend Design	
----------------	---	---

Table of Contents

Abstract.....	2
Meet the team:	3
Member Contribution:.....	3
Executive summary	5
Workflow of Project Activities	6
Requirements	8
State of the Art	10
UX Design Solution	11
Technological Solution.....	15
Value Proposition:	19
Discussion and Future Work	19
Bibliography:	20

Executive summary

This project aspires to develop a children supporting system called Reading Trainer Bobby with the goal of targeting children, having difficulty in reading by using the new AI technology of synthesizing social stories. The fundamental focus of this venture is the enhancement of young people's reading abilities and content focus through personal and creative solutions.

The social stories that are written by the Reading Trainer Bobby for the children are developed using OpenAI's artificial language models, which change according to the reading abilities of the child. It assists in identifying certain learning difficulties and makes reading fun and far more efficient. The key characteristics of the proposed system include story-based generation components that aim to take into account the child's progress, game-based components to involve the users, and adaptable content in order to address progressive learning stages of the child.

The project has evidenced that individual social stories can greatly improve the level of interest and understanding of consumables for children with reading ailments. In this case, the Reading Trainer Bobby presents relevant and timely information that can easily be expanded according to the child's learning progression.

Therefore, Reading Trainer Bobby can be seen as an important improvement in the sphere of educational technologies and the discovering of new tools for practicing reading. Not only does it actively work to solve the important issues related to children with reading difficulties, but also it can significantly revolutionize reading education by using AI content creation. Possible future developments are the addition of more games and a link to other related teaching utilities to enhance the learnin

Workflow of Project Activities

1. Initial Planning and Concept Development

- **Objective:** Describe the project and its intended accomplishments within an organization.
- **Activities:** Defined the project goals, selected the target audience (children with reading difficulties), and outlined the features to be implemented.
- **Outcome:** Developed the initial conceptual design and set project standards.

2. Design and Prototyping

- **Objective:** Create a prototype to test the framework and obtain tangible results for story generation.
- **Activities:** Developed a sample website to simulate user interactions and experiment with AI story creation.
- **Outcome:** Produced a working prototype capable of generating AI-tailored stories for individual users.

3. OpenAI Integration

- **Objective:** Incorporate AI functionality for generating stories.
- **Activities:** Implemented OpenAI's API in the mock website to enable story creation based on user-specific input.
- **Outcome:** Enabled the website to create unique, AI-generated stories tailored to individual users.

4. User Authentication Implementation

- **Objective:** Restrict story generation to authorized users (therapists).
- **Activities:** Implemented registration and login systems to ensure that only therapists with accounts could access the story generation features.
- **Outcome:** Established a secure authentication system for accessing narrative functionalities.

5. JSON Server Setup

- **Objective:** Manage and store generated stories efficiently.
- **Activities:** Developed a JSON server to handle story storage and retrieval for the web app.
- **Outcome:** Created an efficient backend system for proper storage and retrieval of stories.

6. Testing and Iteration

- **Objective:** Verify that system functions and user experience meet expectations.
- **Activities:** Tested the integrated features, focusing on authentication, story generation, and retrieval.
- **Outcome:** Refined the system addressing issues and optimizing performance.

7. Adding Image Generation

- **Objective:** Implement image generation with the story generation
- **Activities:** Tested image generation and used the generated story to create an image according to the story.
- **Outcome:** Delivered a story and image generation system

8. Fetching Stories and Images

- **Objective:** Be able to retrieve the stories content and images and provide them to the webapp
- **Activities:** Developed a fetching system that allows the integrate the stories and images into the web app
- **Outcome:** Created an efficient integration of the contents of the stories into the design

9. Design of the UI

- **Objective:** Prepare the webapp to be user-friendly
- **Activities:** Designed the user interface so the user can use the web app intuitively

-
- **Outcome:** Simple design but intuitive for the user.

10. Final Development and Implementation

- **Objective:** Prepare the project for deployment.
- **Activities:** Finalized the web application by integrating all features, including authentication and story management.
- **Outcome:** Delivered a fully operational system for therapists to create and manage personalized stories.

Requirements

Main Target Groups/STAKEHOLDERS

- **Primary Stakeholders:**
 - **Children with Reading Difficulties:** The target audience who will use Reading Trainer Bobby to improve their reading abilities.
 - **Therapists:** Users who will manage accounts, generate and customize stories for individual children, and provide the tool to the kids.

NEEDS OF THE STAKEHOLDERS

- **Children with Reading Difficulties:**
 - **Engaging Content:** Need content that is engaging and motivating, ensuring consistent practice and interest during sessions.
 - **Personalized Learning:** Require stories tailored to their reading level and specific learning disabilities to address their unique needs.
 - **Interactive Features:** Need interactive elements that enhance interest and make it easier to understand and engage with the material.
- **Therapists:**

-
- **Efficient Story Generation:** Need a tool that enables the creation of a large number of personalized stories efficiently for each child.
 - **User Management:** Require a system for managing user accounts and tracking the progress of multiple users.
 - **Data Storage and Retrieval:** Need an effective method for archiving and retrieving generated stories.

CONTEXT

- **Educational Settings:** The tool will be used in environments such as schools, reading clinics, and therapy centers where children with reading difficulties receive support.

CONSTRAINTS

- **Technological Limitations:** Dependent on the availability and functionality of OpenAI's API, as well as a stable internet connection.
- **User Accessibility:** The system must be user-friendly and accessible to individuals with varying levels of technological proficiency.

GOALS

- **Personalized Story Generation:** Develop a system that generates tailored social stories to address specific reading problems, enhancing the child's learning experience.
- **Secure User Authentication:** Implement robust security measures to restrict access to authorized therapists only.
- **Efficient Data Management:** Create a backend system for the effective storage, archiving, and retrieval of generated stories.
- **User Engagement:** Design engaging content to maintain the child's interest and support the development of their reading skills.
- **Scalability:** Ensure the system can expand to accommodate more users and stories as the project grows.

State of the Art

Read&Write

- Read&Write is a comprehensive software designed to support individuals with reading and writing difficulties. It offers text-to-speech, speech-to-text, and word prediction features to aid users in overcoming reading barriers.
- **Relevance:** Provides tools for enhancing reading and writing skills but does not focus specifically on generating personalized content.

Storybird

- Storybird allows users to create visual stories by selecting artwork and writing accompanying text. It focuses on providing a platform for storytelling rather than addressing specific learning difficulties.
- **Relevance:** While it provides a platform for creative storytelling, it does not incorporate AI for personalized content generation or specifically target reading difficulties.

Ghotit Real Writer

- Ghotit Real Writer is a writing assistant tool for people with dyslexia and dysgraphia. It offers advanced spell-checking, grammar correction, and contextual writing suggestions.
- **Relevance:** Targets writing difficulties and provides assistive technology, though it does not offer story generation capabilities.

UX Design Solution

General Approach

The **Reading Trainer Bobby** is designed as a web-based application with a **Graphical User Interface (GUI)** to provide a user-friendly experience for both therapists and children. The main mode of interaction is **conversational AI** supported by OpenAI to create social tales based on users' particularities. The above outlined system seeks to solve the fractured reading problems of children through developing and providing relevant and appealing storylines that help to improve the reading exercise and understanding in the kids.

Interaction Paradigm

- **GUI-Based Interaction:** Users interact with a visually intuitive web interface that allows for easy navigation and story generation.
- **Conversational AI:** The application integrates OpenAI's API to generate stories based on user input, creating a personalized experience for each child.
- **Personalized Content:** These social stories are then created based on the reading and functional level of the child.

Main Functionalities/Tasks

1. Story Generation:

- Therapists can input specific details about the child's reading level and preferences.
- The system generates a personalized story that addresses the child's needs.

2. Story Storage and Retrieval:

- Generated stories are stored in a JSON server.
- Therapists can retrieve and review stories.

User Workflow(s)

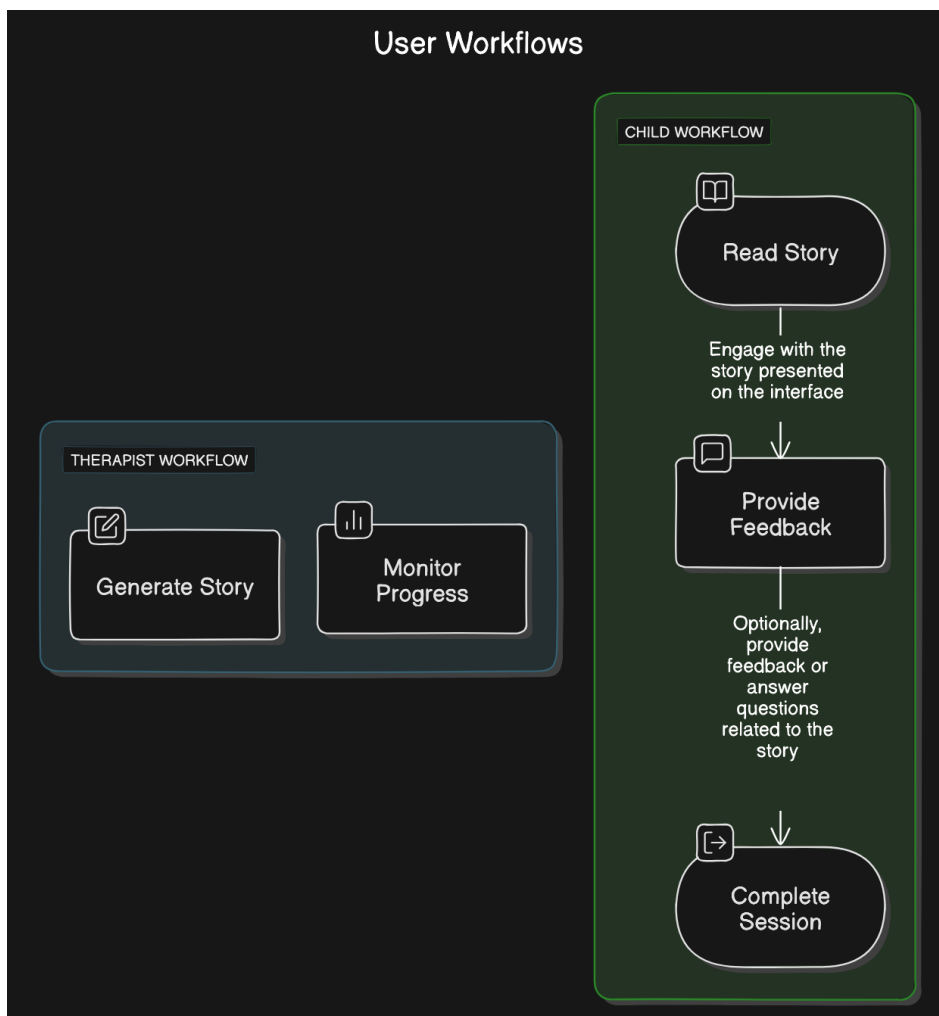
- **Therapist Workflow:**

1. **Login/Registration:** Access the system through secure login.
2. **Generate Story:** Input details and generate a personalized story.
3. **Save/Store Story:** Save stories in the system for future use.
4. **Monitor Progress:** Track the child's progress and review their interaction with stories.

- **Child Workflow:**

1. **Read Story:** Engage with the story presented on the interface.
2. **Provide Feedback:** Optionally, provide feedback or answer questions related to the story.
3. **Complete Session:** End the reading session and log out.

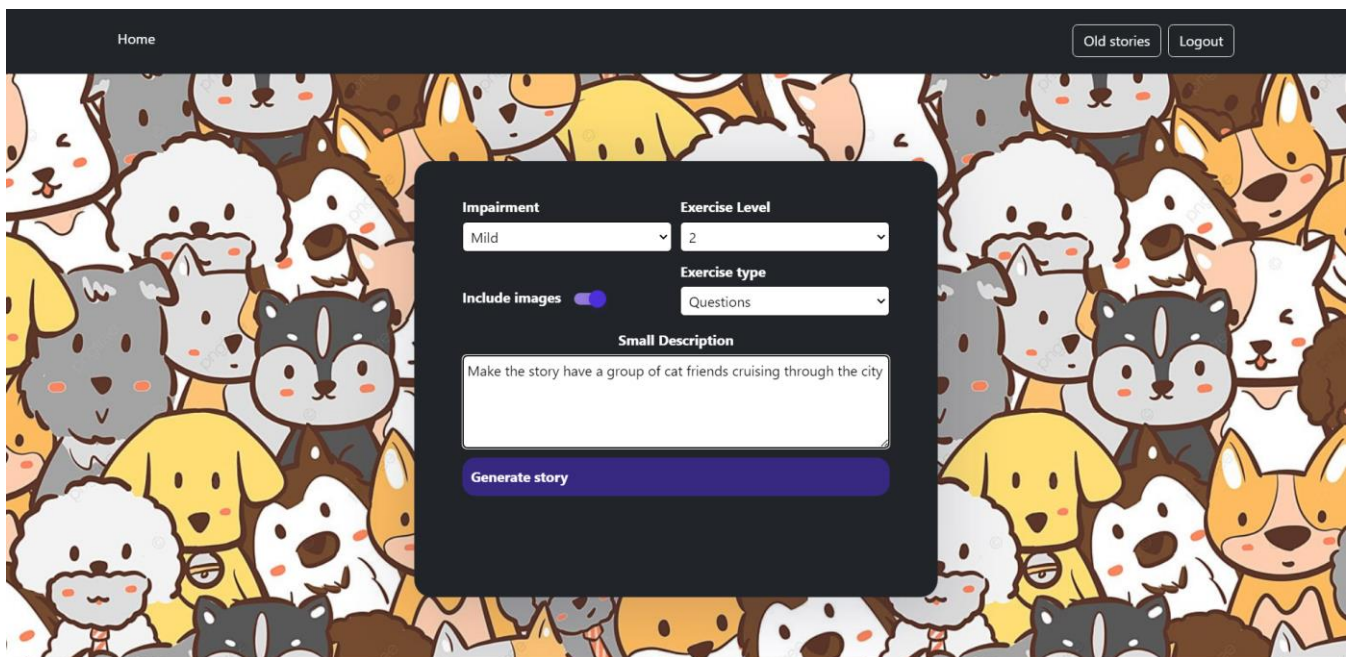
User Workflow Diagram:



Relevant Scenarios

1. Scenario 1: Therapist Generating a Story

- **Description:** The therapist logs into the system and generates a story inputting specific details (e.g., reading level, interests).
- **User Actions:** Therapist accesses the web app, inputs details, reviews the generated story, the story is saved automatically.

The screenshot shows a web application interface for generating a story. The background is a colorful, cartoonish pattern of various animal faces, primarily dogs and cats. At the top, there is a dark navigation bar with the word "Home" on the left and two buttons, "Old stories" and "Logout", on the right. In the center, there is a dark modal form. The form contains several input fields: "Impairment" with a dropdown menu showing "Mild", "Exercise Level" with a dropdown menu showing "2", "Include images" with a toggle switch that is currently turned on, and "Exercise type" with a dropdown menu showing "Questions". Below these fields is a text area labeled "Small Description" containing the text "Make the story have a group of cat friends cruising through the city". At the bottom of the form is a purple button labeled "Generate story".

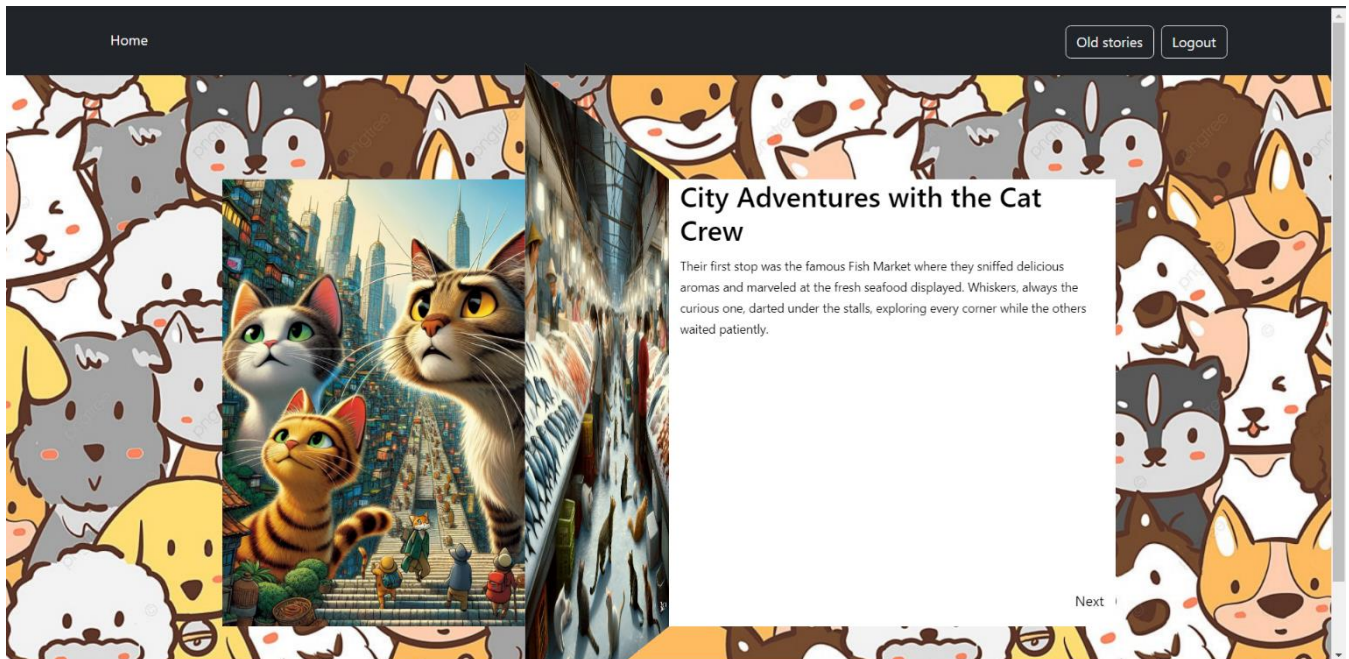
2. Scenario 2: Child Reading a Story

- **Description:** The child reads the story provided by the therapist and answers the questions if there are any
- **User Actions:** Child engages with the content, and provides feedback.

3. Scenario 3: Therapist Reviewing Progress

- **Description:** The therapist accesses the progress report of a child, reviews the history of generated stories, and makes adjustments as needed based on the child's performance.

- **User Actions:** Therapist logs in, accesses the child's profile, reviews progress data, and adjusts story settings or generates new stories.



Motivations for Design Choices

- **GUI-Based Interaction:** Chosen for its ease of use and accessibility, ensuring both therapists and children can navigate the system effortlessly.
- **Personalized Content Generation:** Essential for addressing the unique reading challenges of each child, providing a more effective and motivating learning experience.

Technological Solution

General Technological Approach

The **Reading Trainer Bobby** project employs a combination of web technologies and AI integration to provide a seamless experience for generating personalized stories for children with reading difficulties. The system leverages a web-based platform for user interaction, OpenAI's API for story generation, and a JSON server for data storage and retrieval.

Commented HW Architecture

Since this project primarily involves web-based technology, there is no specialized hardware architecture beyond standard server and client devices. However, here's a general overview:

- **Client Devices:**

Devices such as desktops, laptops, tablets, and smartphones used by therapists and children to interact with the web application. Must have a modern web browser and an internet connection.

- **Server Infrastructure:**

Hosts the web application, manages user authentication, story generation requests, and data storage. For purpose of simplicity it has been done locally

Commented SW Architecture

- **Functional Level:**

- **Frontend:**

- **Technologies Used:** JavaScript, Vue.js, HTML&CSS and Python

- **Backend:**

- **Technologies Used:** Node.js, Json Database and firebase

- **Implementation Level:**

- **Web Server:**

- **Local server**

- **AI Integration:**

OpenAI API and OpenAI Assistants API used for generating personalized stories based on user input.

- **Data Storage:**

JSON server (mock database) which stores and retrieves generated stories and user data in a structured format.

Prompt Engineering Methods

- **General Approach:**

- **Prompt Design:** Prompts are crafted to elicit specific types of content based on the child's reading level and preferences. These prompts guide the AI to generate relevant and engaging stories.

The prompt used is the following:

```
You are Bobby, a friendly assistant specializing in creating reading exercises for children with various reading impairments. You have a set of guidelines and exercises tailored to different levels of impairments. When responding, the only output should be the JSON object.
```

Before generating an exercise, you will receive input in the following format:

```
{  
  "impairmentLevel": "Moderate",  
  "exerciseLevel": "1",  
  "includeImages": true,  
  "exerciseType": "Story",  
  "smallDescription": "Johnny the rabbit"  
}
```

- 'impairmentLevel' indicates the severity of the user's reading impairment.
- 'exerciseLevel' denotes the difficulty level of the exercise (1 being the easiest and 4 being the hardest).
- 'includeImages' specifies if the exercises or stories need to include images. Each paragraph need to have its own image, no more than 1 image per paragraph.
- 'exerciseType' specifies the type of exercise (it can be "Story" for a simple story or "Questions" for a story followed by questions).
- 'smallDescription' is a small description provided to help you create the story.

The prompt clearly defines the role of the assistant as "Bobby," who specializes in creating reading exercises for children with reading impairments. This helps to establish the context and scope of the task.

The prompt specifies the format of the input, including fields like `impairmentLevel`, `exerciseLevel`, `includeImages`, `exerciseType`, and `smallDescription`. This ensures that the input requirements are understood and that the assistant can generate appropriate exercises based on these parameters.

Instructions:

1. **"Exercise Creation"**: Based on the input, create an exercise tailored to the impairment level, exercise level, and exercise type.
 - If the 'exerciseType' is "Story", create a simple story.
 - If the 'exerciseType' is "Questions", create a story followed by questions.
2. **"Story and Content"**: Include a story that matches the small description and is appropriate for the specified exercise level. The content should be exactly 3 paragraphs and each title should be new and unique, you cannot repeat titles. Each id is unique.
3. **"Images"**: If 'includeImages' is 'True', generate and include relevant images in the exercise. Each image should have a URL and a brief description. The images should be directly related to the story content you create.

Image Generation:

1. **"include Image URLs and Descriptions"**: Do not fill this section, is should be empty

Output Schema Example:

```
{
  "id": "Exercise Id",
  "title": "Exercise Title",
  "content": [
    "Exercise content goes here.",
    "You can include questions."
  ],
  "images": []
}
```

Example Output:

For a "Story" exercise:

```
{
  "id": "T74c",
  "title": "Johnny the Rabbit's Adventure",
  "content": [
    "Johnny the Rabbit lived in a lush green forest...",
  ],
  "images": []
}
```

For a "Questions" exercise:

```
{
  "id": "T74c",
  "title": "Johnny the Rabbit's Adventure",
  "content": [
    "Johnny the Rabbit lived in a lush green forest...",
    "What did Johnny find in the forest?"
  ],
  "images": []
}
```

Your task is to create a JSON output with the exercise details based on the provided input. Here are example outputs. Use the examples to generate the stories but without repeating them, use the same structure only changing themes and characters in the story. Each story must have 4 paragraphs.

The prompt provides clear instructions on how to create the exercise, including tailoring it to the impairment level, exercise level, and exercise type. This helps in generating appropriate content.

It also specifies that the output should be a JSON object, with a clear schema that includes fields for id, title, content, and images. This structured approach ensures that the output is consistent and easy to process.

It emphasizes not repeating stories and generating new themes and characters for each exercise. This encourages creativity and ensures that each exercise is unique.

```
{
  "id": "",
  "title": "The Missing Medal",
  "content": [
    "In the garden, the sun is shining bright. The monkey Max swings among the tall banana trees. In the garden, there are many colorful flowers. Max finds a shiny gold medal and starts to play with it. But suddenly, the medal slips from Max's hand and lands in a patch of bushes! Poor Max is very upset. Monkey Max searches for his medal everywhere, peeking under the bushes. Next to a bush is a tiny ant. Next to the ant is a delicate butterfly. The ant waves at the butterfly. Max Monkey says hello to them. "Have you seen my gold medal?" asks Max. But the butterfly doesn't answer and flutters away. A rabbit and a hedgehog are nearby. The rabbit wears a green vest. The hedgehog is carrying a little basket. Max Monkey waves at them.",
    "The rabbit asks, "Hello, who are you? What are you looking for in the garden?" Max replies, "I'm Max! Have you seen my gold medal?" The rabbit says, "No, sorry." So Max continues his search. Monkey Max hops and hops through the garden and arrives at a small pond. Max sees a frog jumping on lily pads. There is also a dragonfly hovering above the water. The frog leaps after the dragonfly. A little girl in a blue dress sees them. The dragonfly is spotted by the little girl. "Hey, hello, little girl! Have you seen my gold medal?" asks Max. The girl gets startled and runs away. Poor Monkey Max feels even lonelier and misses his gold medal. Suddenly, he hears a rustling. A baby deer and a raccoon emerge from the bushes. The raccoon is carrying some berries. Both of them sit on a log to rest. The baby deer is chased playfully by the raccoon.",
    "Max Monkey stops them. But the raccoon runs off. Max feels increasingly sad and lonely. A bluebird is perched on a branch. A squirrel gathers nuts on the ground. A turtle crawls slowly by. The bluebird sings a cheerful song. The squirrel says, "Hey, Max, I have an idea! Why don't we all search for your gold medal together? I'll check the trees, the bluebird can look from above, and the turtle can search the ground." What a wonderful idea the squirrel had! The friends are ready to start searching! The squirrel is carried by Max. The bluebird soars above. The friends are eager to explore! Everyone heads out to find the lost medal. Max is gently lifted by the bluebird. The squirrel keeps an eye out below. The friends search and search, until they get tired and decide to rest by the pond.",
    "In the morning, a rooster's crow wakes them up. Max is stirred by the rooster's call. The squirrel scolds the rooster. "What a noise! It's only 6 a.m.!" The squirrel exclaims. The rooster chuckles. "Friends, today, there's a big picnic with all the garden animals!" A deer sits next to Max and the rooster. A bear with a picnic basket also arrives. A duck quacks excitedly. The squirrel helps the duck. A goose waddles over with a fruit pie. The rooster is jostled by the bear. The goose watches them. "What a delicious pie! I love fruit pies!" Max exclaims. Max eats, plays, and dances all day with the garden animals. He is so happy that he forgets about the gold medal. As the day ends, he decides to rest by the pond. Monkey Max is joined by the squirrel. Max and the squirrel relax by the water's edge. As they sip the cool water, the squirrel notices something shiny under the water. Max is called over by the squirrel. A fish swims by, looking curious. "It's my gold medal!" Max exclaims. The squirrel is lifted up by Max. All the animals cheer with joy! Yay! Max has finally found his gold medal! The music starts again, and everyone dances joyfully.",
  ],
  "images": []
}
```

Providing example outputs helps to clarify the expected format and content for both "Story" and "Questions" exercises. This serves as a useful reference for generating new exercises.

Value Proposition:

The Reading Trainer Bobby is a perfect solution applying the concept of artificial intelligence to develop children's reading ability in the case of reading difficulties. The platform's value lies in its ability to provide:

Personalized Story Generation: This way, the system, with the help of OpenAI's API, generates the specific social stories for individual children with reference to the assessment of their reading level and several peculiarities. This personalization strengthens interaction and enhances learning since it is done based on the individual needs of the learners.

Accessible and Interactive Learning: The developed and presented interface of the application makes comprehension of the reading practice easy and fun for the therapist as well as the children.

Discussion and Future Work

The development of Reading Trainer Bobby has highlighted several strengths and challenges:

Strengths:

Innovative Use of AI: Thanks to the OpenAI's API, it became possible to create unique content distinguished from simple traditional texts.

User-Centric Design: With regards to the child and therapist, accessibility is prioritized and they are able to interact easily and naturally while using to the platform.

Challenges and Critical Aspects:

AI Limitations: Native articles produced from AI inputs are specific because they reflect individual prompt, but the quality of the articles' content greatly depends on the prompt provided. Ensuring that prompts consistently produce appropriate and engaging stories is a key challenge.

Future Directions:

Short to Medium Term:

Enhance AI Capabilities: Enhance the features about prompt engineering and consider other possibilities of extended functionality of AI, including its further learning from the users' feedback.

Expand Content Library: Add more types of stories and subjects that would interest more audiences and for all ages and levels of difficult.

Improve Design: Add design features which allows the user to do more functionalities such as filtering old generated stories.

Make user profiles: Give the therapist the ability to create profiles for the childrens he is supervising with task, progress reports, etc

Long Term:

Multimodal Learning: Gather information on the use of additional media aspects like audio and additional graphics to supplement the already existing textual content to make text-to-speech or speech-to-text exercises.

Scalability and Global Reach: To expand the application of the tool, create versions in multiple languages to increase reach.

Bibliography:

Works Cited

codingstar-jason. "3D-Book-Tutorial-Basic-CodingStar/README.md at Main · Codingstar-Jason/3D-Book-Tutorial-Basic-CodingStar." *GitHub*, 2021, github.com/codingstar-jason/3D-Book-Tutorial-Basic-CodingStar/blob/main/README.md. Accessed 3 Aug. 2024.

“Ghotit Dyslexia Real Writer & Reader.” *Dyslexia Writing & Reading Assistive Technology | Ghotit Dyslexia*, www.ghotit.com/dyslexia-software-real-writer-for-windows.

iamshaunjp. “GitHub - iamshaunjp/OpenAI-Basics at Lesson-3.” *GitHub*, 2023, github.com/iamshaunjp/openAI-basics/tree/lesson-3. Accessed 3 Aug. 2024.

---. “Vue-3-Firebase/Ninja-Jobs at Lesson-56 · iamshaunjp/Vue-3-Firebase.” *GitHub*, 2020, github.com/iamshaunjp/Vue-3-Firebase/tree/lesson-56/ninja-jobs/. Accessed 3 Aug. 2024.

leelanasimha. “Vue3-Course/Vue-Authentication at Master · Leelanasimha/Vue3-Course.” *GitHub*, 2020, github.com/leelanasimha/vue3-course/tree/master/vue-authentication. Accessed 3 Aug. 2024.

Net Ninja. “Net Ninja.” *YouTube*, 2024, www.youtube.com/watch?v=7iDGJolHFmU. Accessed 3 Aug. 2024.

“Read&Write for Education - Reading, Literacy & Assistive Software.” *Texthelp*, 2021, www.texthelp.com/products/read-and-write-education/.

Reading Trainer - Improve Your Reading Power. readingtrainer.com/.

scalablescripsts. “GitHub - Scalablescripsts/Vue-Auth.” *GitHub*, 2021, github.com/scalablescripsts/vue-auth. Accessed 3 Aug. 2024.

Storybird. “Storybird - Read, Write, Discover, and Share the Books You’ll Always Remember.” *Storybird*, 2019, storybird.com/.

Tutorialss, Coding . “CSS FLIP BOOK on CLICK USING HTML and CSS ONLY | CREATIVE FLIP BOOK EFFECT.” *Www.youtube.com*, www.youtube.com/watch?v=A5fdu8HZKic. Accessed 22 July 2021.

VIPOND, ANTHONY. “Flipbook.” *Winterwind.com*, 2018, www.winterwind.com/tutorials/css/3. Accessed 3 Aug. 2024.