

README

06 July 2024 10:11

e-KG v.2.0

License

****This project is licensed under the MIT License. See the LICENSE file for more details.****

Credits

Developed by Ashutosh Singh under the guidance of Mr. Maheshwar Dev.

The first version was developed in collaboration with my classmates when we were in 8th grade.

Special mentions: Divij Gupta, Aditya Dabgotra, Nitish Saini, and Himanshu S. Jha.

Disclaimer

The developer takes no responsibility for any damage or issues that may arise from the use of this project. Use it at your own risk.

Previous Version

Unfortunately, the initial version of the ****e-KG**** project was lost due to a ransomware attack. This earlier version of the project did not work as a single file; instead, it utilized file handling to manage and display multiple files together whenever needed. While this approach had its benefits, the current version has been redesigned to function seamlessly as a single file, ensuring easier access and usability.

Description

****e-KG**** is a Scratch project designed to animate characters, including alphabets and numbers, on the screen. This project serves as a visual and interactive way to display various characters through smooth animations.

Features

- Displays animations of alphabet characters (both uppercase and lowercase).
- Displays animations of numbers.
- Simple and intuitive design.
- Documentation explaining the design and functionality of the project.

Installation and Usage

1. Download and open the project:

- Download the e-KG project file from the repository.
- Open the project in Scratch 3.0 or any compatible version.

2. Run the project:

- Click the green flag icon to start the animation.
- Watch as the characters are animated on the screen.

3. ****Easy step**** :

- Just go to <https://astebe80.github.io/e-KG/>, and you can run the program online.

Documentation

For a detailed explanation of the design and functionality of the project, refer to the included documentation file named "Readme_e-KG_v.2.0.pdf". The documentation covers the following sections:

- Introduction: Overview of the e-KG project.
- Design: Explanation of the design choices and structure of the project.

- **Functionality:** Detailed description of how the project works.
- **Rough Work:** Contains notes and rough work, which can be ignored if not needed.

Contributing

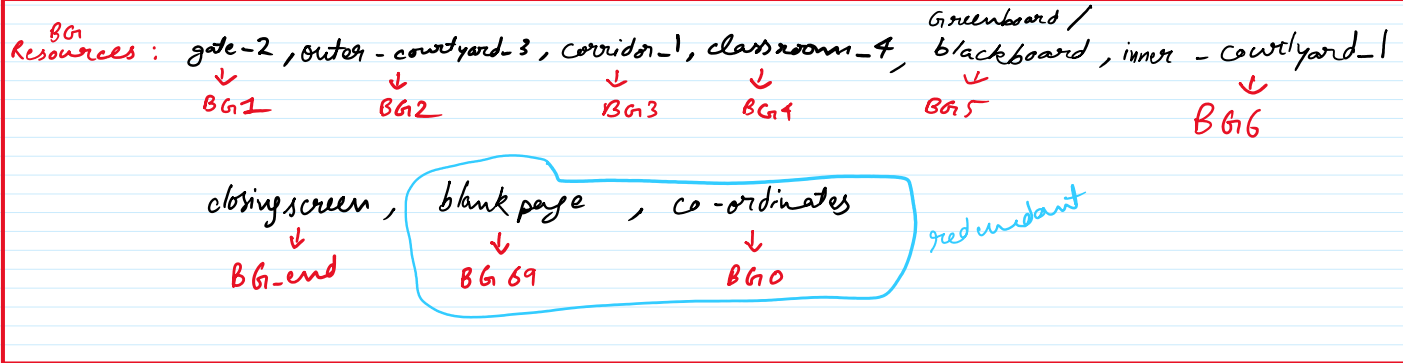
Contributions to the e-KG project are welcome. Feel free to fork the repository and submit a pull request if you have any ideas for improvements or new features. Make sure to follow the contributing guidelines.

Contact

For any questions or feedback, please contact me at astebe80@gmail.com.

★ Backdrops

→



★ Voicepacks

→

```

** audio source - https://ttsfree.com/
49. English (UK) - Sonia - 0% pitch - 0% adjusted voice speed
limit - 50 conversions/day

```

Can either use this or TTS built-in Scratch 3.

Update: - TTS is unreliable, so I will be sourcing local files which I generated from the above listed site.

★ Notes

→

```

** I have yet to learn GIT and managing project on GitHub
1) Version control - GIT
2) Project hosting - GitHub ( private initially, then I'll make it public)

** Documentation takes the least priority. I'll do it at the end.
It will include details of the project, difficulties faced, respective solutions.
The usual README.txt file will also be included at later stages.

Updates will be managed through Git.

Will try to recreate it using Django and host it on GitHub.

```

TTS is unreliable indeed, started working again. I'll try to switch to sourcing local files in the later versions.

★ Packaging

```

https://packager.turbowarp.org/
convert it into either of the formats present on the site

```

★ System Requirements

→

```

PC
OS:
Windows 10 version 14316.0 or higher
Architecture:
x64
Developed by
Scratch Foundation & MIT Media Lab

```

→ Also, instead of building project on Scratch, I'm using TurboWarp Desktop for development.

→ It has much more features than Scratch, while being an extension of Scratch itself.

→ Auto packaging, auto-saves & much more.

P.S., No I'm not being sponsored.

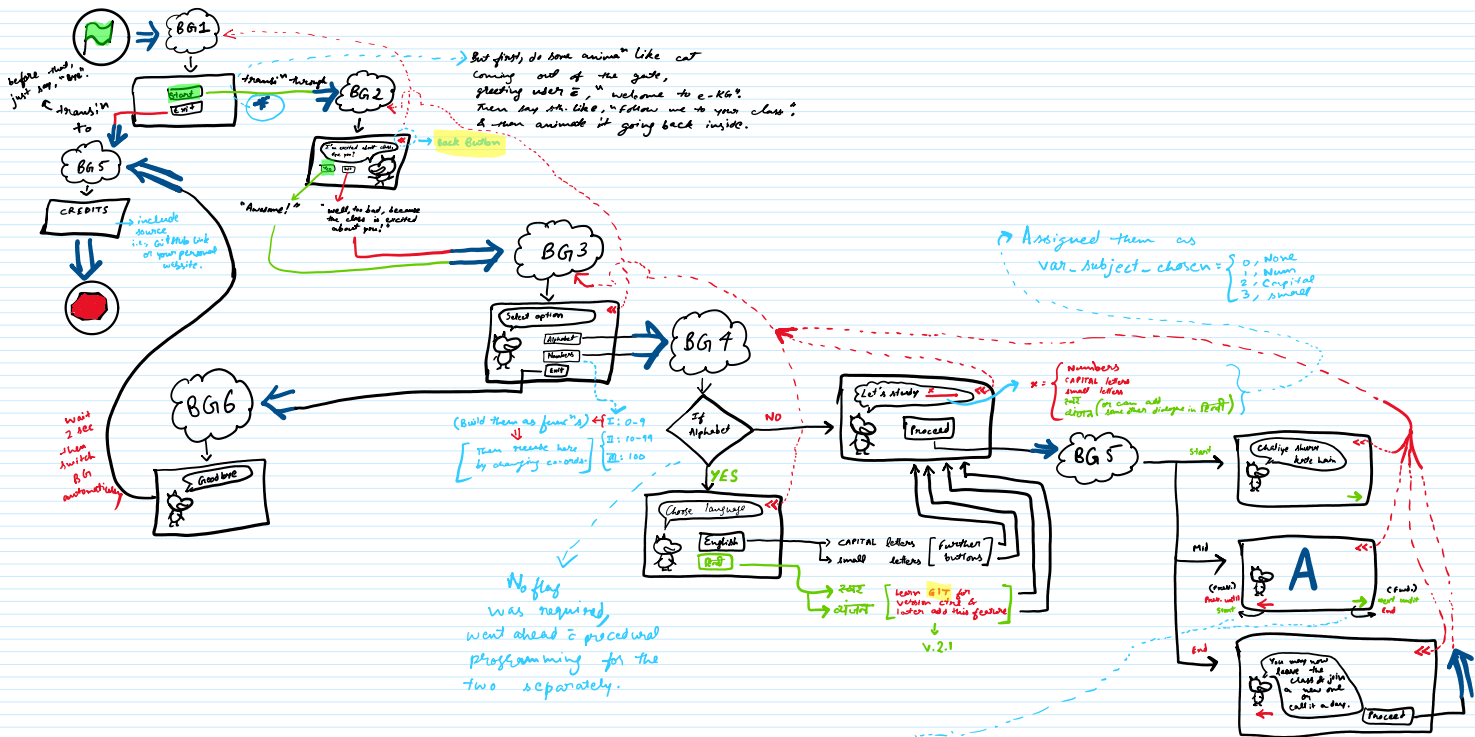
★ Sprites

- Pen/Chalk ✓
 - Professor Scratch ✓
 - Back button << ✓
 - Prev. button ✓
 - Next button ✓
 - Buttons
 - Start ✓
 - Exit (x2) BG 1,3 ✓
 - Proceed (x2) BG 4,5 ✓
 - Yes ✓
 - No ✓
 - Alphabet [Flags] ✓
 - Numbers [Flags] ✓
 - English ✓
 - Pen ✓
 - CAPITAL ✓
 - small ✓
 - ۲۰۲۲ ✓
 - ۲۰۲۱ ✓
- values for var. 'x' ✓

→ I also included one 'stop' sprite s.t. when pen touches it, it stops drawing. But that was more complex solⁿ. Setting up the 'stop' = fwd & prev did the job perfectly fine.

Design Overview

V.2.0



These were giving me a hard time... depends steps over the problem... write up & get it done!!
Tip - Step over it.

initial pts	2PC (Open process communication)
0, 150, 20 1 [var_subject_chosen]	when added, broadcast msg, change pts_state by +1 when receives msg, change pts_state by -1
0-1 [var_subject_chosen]	when deleted, broadcast msg, change pts_state by +1 when receives msg, change pts_state by -1

★ Test programs for features

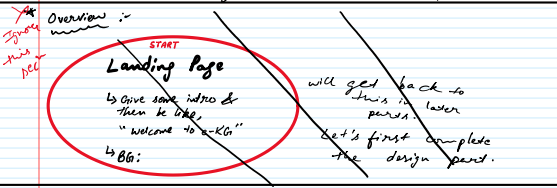
- Animation ✓
- Buttons & loops (back, fwd, etc.) ✓
- GIT for version control ✓
- After packaging, (Refer to 'required resources'), deployment of .htm, .exe & .sb3 files on GitHub repository. ✓

//ROUGH WORK

Scratch :- Project Revival

⇒ Note :- This is v.2.0., early v1.0 was lost to transamware attack.

- * Aim :- To demonstrate implementatⁿ of Scratch 2.0.
- * Course of Act :- Rebuilding e-KG Ltd. interactive app.



Design

Resources :- gate-2, inner_courtyard-3, corridor-1, classroom-4, Greenboard/blackboard, outer_courtyard-1

BG1 BG2 BG3 BG4 BG5 BG6

