

OurProject

Update ver. 0.4 (14.10.24)

-Код, связанный с нейросетью, был доработан для требующихся задач, закомментирован, выгружен на гитхаб.

-Был найден пробный вариант 3д модели аватара для тестирования остального функционала.

-Был проведен ресерчинг на тему соединения нейросети с 3д моделью, попытки это сделать пока что не увенчались успехом.

-Интерфейс приложения и визуал:

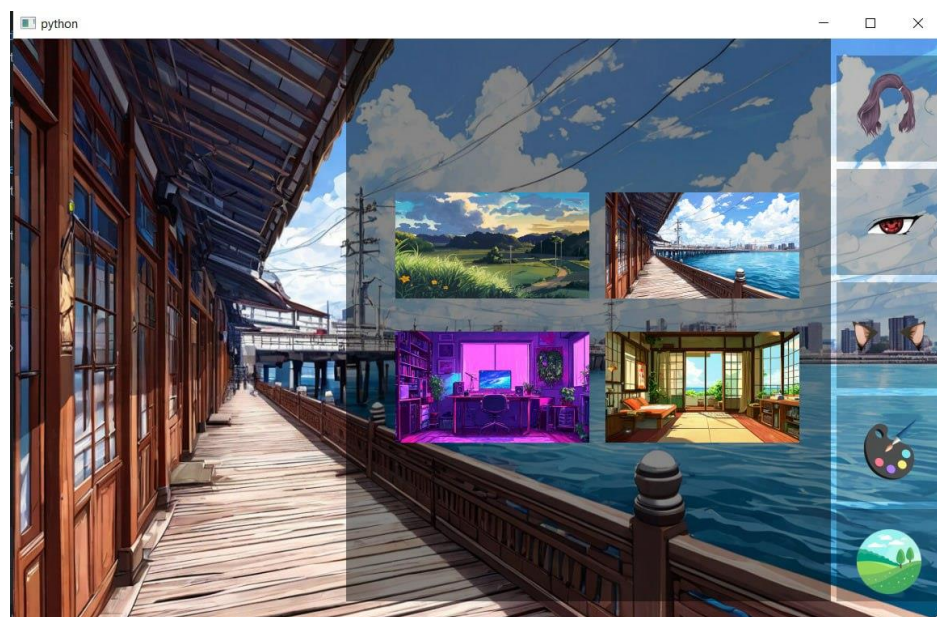
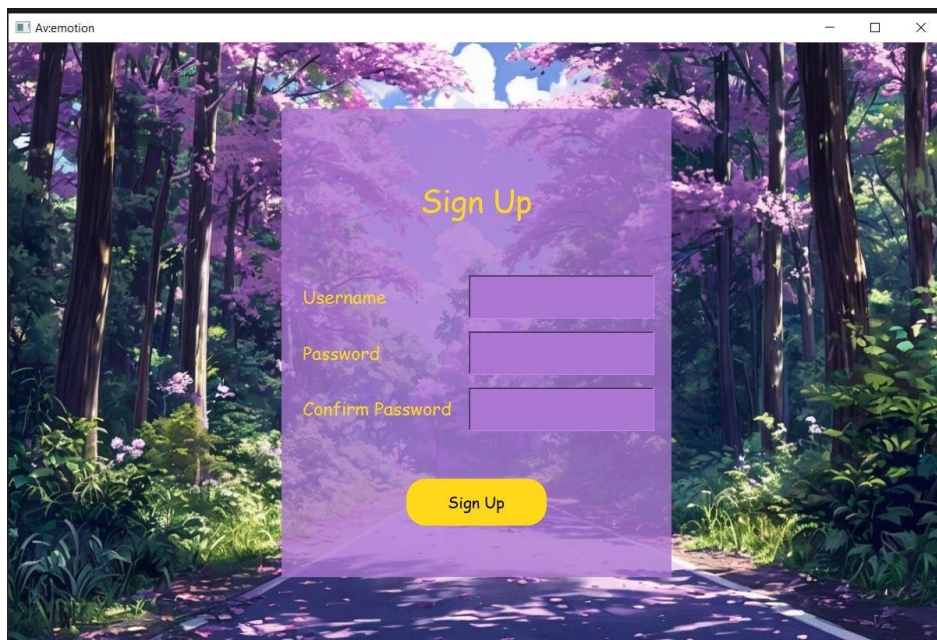
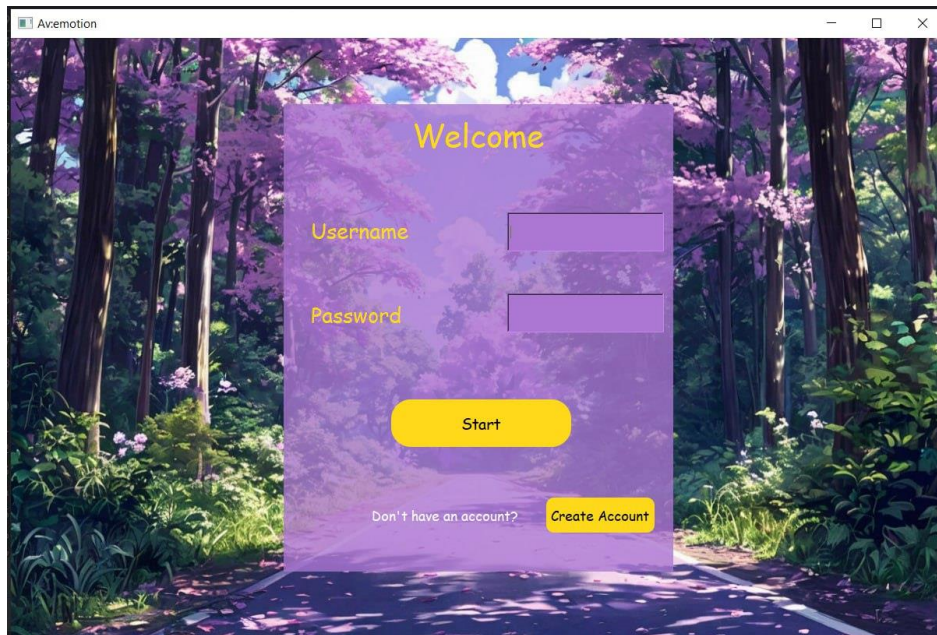
- Была реализована локальная БД в PhpMyAdmin с помощью сервера MAMP

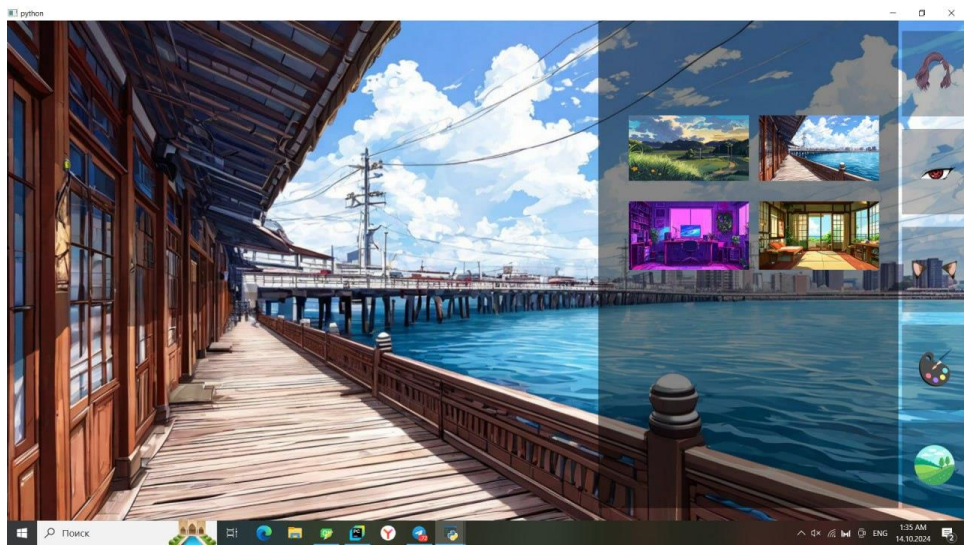
#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	ID	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
2	username	varchar(60)	utf8_general_ci		Нет	Нет			Изменить Удалить Ещё
3	password	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить Ещё
4	created_at	timestamp			Нет	CURRENT_TIMESTAMP			Изменить Удалить Ещё

- Был создан класс Database для управления подключением к БД MySQL

ID	username	password	created_at
1	username	\$2b\$12\$bQH6Zh1qNZi2LNdKb.Z9aO8grcgkZGWUYhJh1hD1x0P...	2024-10-06 15:49:09
2	wlore	\$2b\$12\$3c1cr2idJoodFgjkGLRjFOv3TDw4gZIJrTzUpQ4y9OJ...	2024-10-06 16:07:25
3	poppo	\$2b\$12\$OvnAvYbUH8G/pkD90lNsVuzZ3QnisHycWuG/l3Ej31G...	2024-10-06 16:13:58

- Реализовано хэширование пароля и добавление пользователей в БД
- Проект был разделен на файлы для удобства в работе с ним
- Теперь после регистрации/авторизации появилась возможность перейти к главному окну нашей программы(шаблону GUI)
- Также были добавлены проверки введенных данных и обработки ошибок при работе с БД
- У окон авторизации и регистрации был сделан итоговый дизайн
- Также была исправлена ошибка с автоматическим изменением размера окон

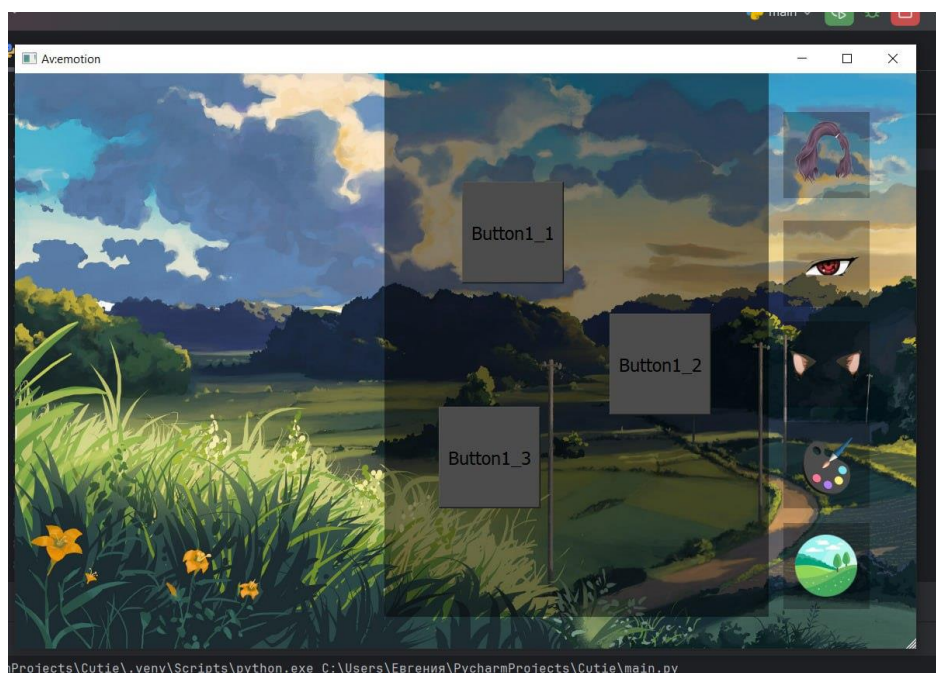
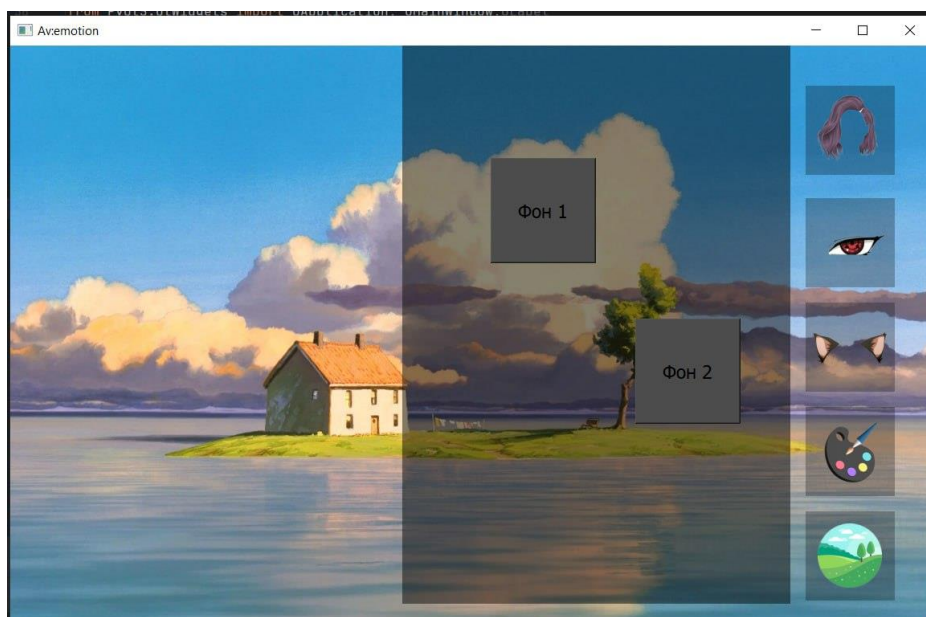




Update ver. 0.3 (30.09.24)

- Удалось подключить 2 нейросети: Deep Face и EmoAffectNet. Обе имеют ряд плюсов и минусов, на данный момент решается, какая именно будет использоваться в дальнейшем.

- Был разработан черновой вариант графического интерфейса для редактирования аватара. И также была прописана его логика посредством кода. В результате, программа создает имеет интерфейс с несколькими страницами и возможностью динамически менять фоновое изображение при нажатии кнопок.



Update ver. 0.2 (23.09.24)

- Для обнаружения лица с камеры в реальном времени был заменен метод Хаара с каскадами на метод поиска лицевых особых точек. Теперь захват лица во время его движения происходит с высокой точностью, помимо этого фиксируются изменения в положении отдельных частей лица таких, как глаза, брови, губы и тд.

- На ресурсе Hugging Face была найдена 1 нейросеть Emo-AffectNet model, которая работает непосредственно с видео. Попытки запустить ее для того, чтобы разобраться в принципах работы, были практически полностью безуспешны, максимальное время работы программы 0,6 сек.

- Визуал:

1. Было создано окно для авторизации Sign up
2. Был прописан код для графического интерфейса (class Login, отвечающий за аутентификацию пользователя, и class Sign up, отвечающий за его регистрацию в нашем приложении)
3. Удалось подключить графический дизайн к прописаному для него коду в PyCharm, запуск программы теперь реализуется через ide
4. Кнопки интерактивные и отвечают за определённые функции
5. Чтобы сделать ввод пароля безопасным, был добавлен метод Echo_mode, для скрытия его символами

Update ver. 0.1 (16.09.24)

- Написан 1 вариант кода, осуществляющий захват видео с камеры в реальном времени, обнаружение лица и выделение его посредством зеленого прямоугольника. (каскады Хаара)

- Найдены варианты нейросетей, которые могут быть использованы для определения эмоций:

1) Библиотека Face expression recognition(FER). Пока что попытки подключить ее к проекту были неудачными;

2) Библиотека DeerpFace. Пока рассматривается как запасной вариант, так как имеет ряд недостатков: не всегда корректно отображает расу и возраст, меняет их в зависимости от выражения лица одного и того же человека;

3) Context-Aware Emotion Recognition (CAER). Пока что тоже рассматривается как запасной вариант.

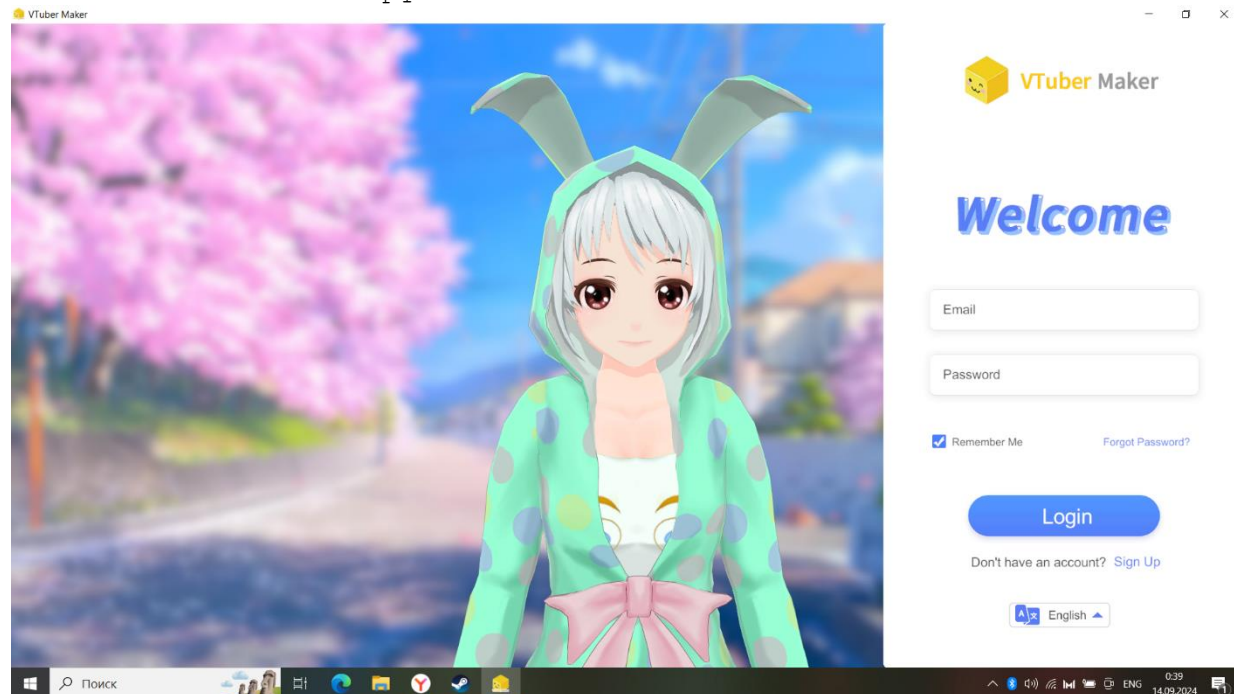
- Ища информацию о том, как сделать 3д аватар с нуля, были сделаны следующие выводы:

1) есть множество приложений/сайтов с созданием аватара, в основании которых идет загрузка своего фото;

2) приложения/игры с созданием аватаров используют уже готовые шаблоны(локации, внешность, одежда).

- Найдены референсы для графического интерфейса:

Vtuber Maker – пример программы, которая имеет схожий функционал с нашей задумкой. В первую очередь будет создан экран авторизации с использованием библиотеки PyQt Питона и инструмента Qt Designer (визуальный редактор, который нужен для удобства создания пользовательского интерфейса).



Пробный пример экрана авторизации, который был создан в процессе изучения редактора

