Здравствуйте, уважаемые члены жюри. Я – Вершинин Сергей и Мой колега – Воробьев Сергей, учащиеся 11 лицейского класса ГБОУ школы №924 хотим представить Вам наш проект SOLVE-GIA. Он находится в процессе разработки и улучшения.

В наши дни сайты для подготовки к экзаменам становятся все более популярными. Они предлагают удобный формат подготовки в любом месте и в любое время. Однако, задания в варианте не имеют единой медианы сложности, что создает проблемы для некоторых учителей и их учеников. Наша команда решила разработать свой собственный сайт с новым, более продвинутым функционалом, который будет учитывать принципы работы существующих сайтов и предлагать дополнительные полезные возможности. Веб-приложение будет иметь удобную динамическую систему сложности, рассчитывающуюся по формуле, затрагивающую количество людей, прошедших тест, правильность выполнения и время выполнения задания. Для этого нам нужно изучить большой объем информации, разработать формулу и логику расчета сложности каждого задания в отдельности, а также продумать дополнительные полезные возможности для нашего веб-приложения.

Проблемы, которые удалось выделить:

1 Неравномерность сложности заданий на сайтах для подготовки к экзаменам.

2 Весьма ограниченный функционал Telegram бота или его отсутствие.

3 Плохая отказоустойчивость из-за отсутствия брокера сообщений.

Цель нашего проекта – создание собственного ядра веб-приложения и телеграмм бота с расширенным функционалом.

Задачи проекта:

1 Поиск и анализ новой информации по теме.

2 Разработка формулы 3. Создание сайта и телеграм бота 4 тестирование и завершение проекта

Разработку нашего проекта мы решили начать с поиска информации и анализа аналогов, после которого мы определили проблемы и начали работу над веб-приложением. Основным языком программирования мы выбрали Python, потому что это не только самый популярный язык программирования, но и очень удобный, имеет много фреймворком и отличное сообщество. В качестве фреймворка для создания backend’a сайта для начала выбрали FLASK, но после некоторого времени разработки мы перешли на Django, потому что он имеет больший функционал, а также встроенные функции, которые на FLASK пришлось бы реализовывать самим. Для разработки многофункционального Бота в Telegram мы решили использовать Aiogram, тк он имеет большое количество встроенный функций, таких FINITIAL STATE MACHINE, а также более удобная логика самого кода.