***“Schoosch”***

***Электронный дневник***

Создано Степаном Чепайкиными

под руководством Михаила Чепайкина

при поддержке Гимназии имени Бориса Пастернака

2022 год

Цели: сделать работу учителей и досуг учеников чуть проще, создав и предоставив приложение-дневник с множеством функций.

**Основные функции**:

1. Для ученика:

а) централизованное расписание

б) возможность смотреть заданное дз и отмечать его выполненность

в) возможность видеть все свои оценки и комментарии к ним

г) возможность оставить отзыв учителю

1. Для учителя:

а) возможность видеть свое расписание на неделю

б) можно посмотреть список всего класса

в) ставить отметки любому ученику и писать ДЗ (для всего класса и личное)

г) возможность видеть все отметки любого ученика за весь год

1. Для наставника:

а) можно посмотреть успеваемость контролируемых учеников

б) возможность отметить выполненность задания учеником

в) возможность следить за выставлением учителями заданий классам

1. Для родителя:

а) возможность увидеть успеваемость всех учащихся детей по отдельности

б) возможность видеть выполняемость учеником дз

1. Карта школы для доступа к легкой навигации
2. Чаты между любыми двумя пользователями приложения для быстрой связи

**Используемые программы:**

Программа для написания кода Visual Studio Code, фреймворк для разработки мобильных приложений Flutter, эмулятор андроид-устройства на компьютере для тестирования и отладки кода.

**Причины и смыслы создания:**

Основная концепция создания этого приложения и всего проекта в целом - это создание единого пространства для учеников, родителей и учителей для решения сразу нескольких проблем, таких как:

1. Дисциплина - распространенная проблема многих школ, связанная с тем что ученики не выполняют дз по ряду причин. Этот проект призван помочь исправить ситуацию.
2. Связь между учителями и учениками - в большинстве случаев связана с тем что ученики не имеют возможности быстро связаться с учителями. Это может привести к незнанию о наличии ДЗ или банально к непонятному материалу.
3. Общая открытая база информации - все данные в приложении синхронизируются и обновляются автоматически чтоб у каждого пользователя, будь то ученик или учитель, был доступ к самому последнему расписанию, дз, отметкам и тд
4. Доступность информации - то, что приложение разработано для использования на любых платформах, дает возможность пользователям получить доступ к интересующей их информации в любой момент с помощью смартфона или ПК.

Приложение создается под впечатлением от крайне удачного внедрения в Москве электронной платформы МЭШ около 8 лет назад. Однако за пределами города эта платформа крайне мало распространена, отчасти поэтому появился этот проект.

Ученик будет иметь возможность посмотреть расписания для своего класса, увидеть все свои оценки за день или за все время учебного года, увидеть свои ДЗ на любой день и отметить их выполнение для самоконтроля, воспользоваться картой школы и даже оставить отзыв любому из учителей - это поможет выявить учителей, чье отношение и/или поведение негативно воспринимается многими, а также решить единичные конфликты типа учитель-ученик.

Учитель будет иметь возможность видеть всех учеников для всех классов для которых преподает, поставить оценку любому из них и добавить к ней комментарий по желанию, задавать ДЗ любому из классов или конкретному ученику как личное ДЗ, а также оценивать успеваемость классов через таблицу оценок всех учеников выбранного класса.

Родитель имеет возможность следить за успеваемостью своих учащихся в подключенной школе детей и проверять выполненность учеником ДЗ в любой день.

Естественно, подключенных детей может быть несколько.

В первую очередь, приложение направлено на учеников и учителей, чтобы повысить качество обучения и удобство преподавания и обойти многие бумажные работы, с этим связанные.

К примеру, оно способно полностью заменить собой дневник ученика - все дз и оценки будут там.

Оно может заменить собой журнал учителя - списки учеников, их оценки и ДЗ будут доступны в одном месте в любой момент.

**Создание:**

Приложение создавалось при участии директора и учителей Гимназии имени Бориса Пастернака, для того чтобы действительно решать проблемы обычных учителей и выполнять все свои функции так как нужно.

В создании были задействованы современные технологические решения для более быстрого и более простого написания логики приложения.

Первым является фреймворк мобильной- и веб-разработки Flutter, представленный компанией Google в 2015 году. Это на данный момент одна из лучших и самых быстроразвивающихся технологий с большим потенциалом.

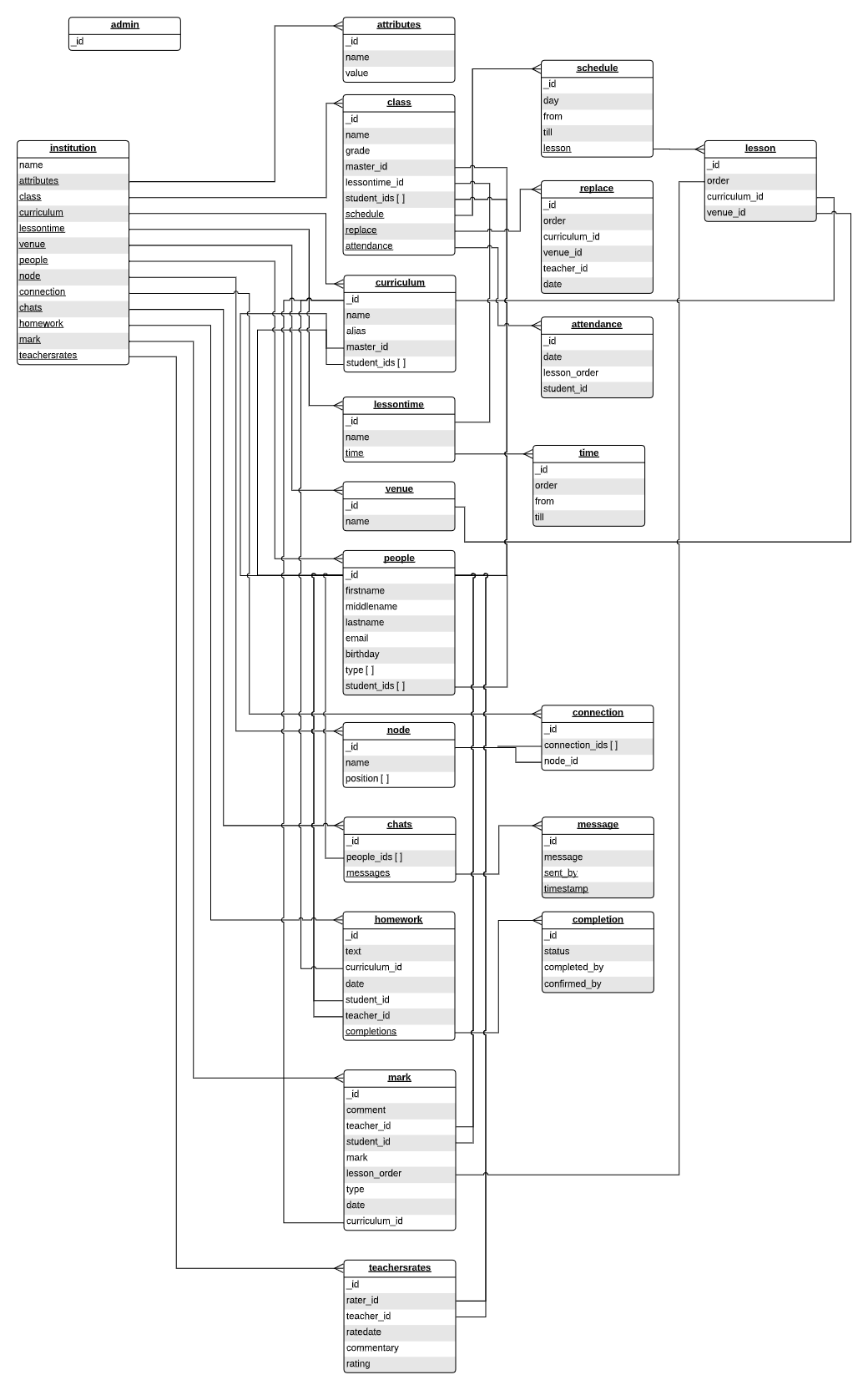
Далее, для хранения и быстрого доступа к данным был использован бэкенд сервис Firebase, также созданный компанией Google. Это позволяет легко объединить эти две технологии в проекте. Firebase сам по себе дает возможность пользоваться их хранилищем данных, а также в любой момент доставать любую информацию из этого хранилища. Он представляет из себя NoSQL базу данных с понятным интерфейсом и множеством дополнительных функций, способных помочь в создании многих проектов.

Также в процессе создания мы часто обращались к сторонним библиотекам и решениям, чтобы не “изобретать велосипед заново”.

Интерфейс мы постарались сделать удобным для любого пользователя. Самые основные функции приложения доступны сразу при загрузке на домашнем экране, чтобы ускорить любое действие, будь то просмотр расписания или выставление новой оценки.

**Документация:**

Для упрощения создания моделей данных в базе и лучшей визуализации связи всех систем, цепочек и атрибутов внутри этих данных была создана обширная документация, которая приведена ниже.



Поле \_id у каждого блока отведено под обозначение локального идентификатора элемента внутри базы.

**INSTITUTION**

Это самый первый и самый главный объект в базе. Собственно, база и представляет из себя список “институтов”, или же учебных заведений. Он объединяет в себе всю информацию о конкретной школе:

1. люди - учителя и учащиеся
2. классы - их ученики и расписание
3. выставленные оценки
4. заданные домашние задания
5. план школы
6. кабинеты школы
7. расписания - действующее и прошлые
8. и тому подобное

Далее каждое из полей будет описано более подробно.

**CLASS**

Объект, содержащий в себе информацию об одном из классов в школе. Содержит атрибуты:

| Название | Описание |
| --- | --- |
| name | Внутреннее для школы обозначение класса, по типу буквы или название |
| grade | Параллель (или же класс) к которому принадлежит класс |
| master\_id | ID человека - классного руководителя класса, если есть |
| lessontime\_id | ID расписания для данного класса, достается и изменяется динамически из другого объекта |
| student\_ids | Список всех ID учеников класса |
| schedule | Коллекция объектов SCHEDULE для класса. Всего их 5, на каждый из учебных дней, хотя конечно их может быть сколько угодно. |
| replace | Коллекция объектов REPLACE у класса, |
| attendance | Коллекция объектов ATTENDANCE у класса |

**SCHEDULE**

Этот объект хранит информацию о расписании для конкретного класса на конкретный день недели. Атрибуты:

| Название | Описание |
| --- | --- |
| day | Цифра текущего дня недели, начиная с понедельника |
| from | Дата, когда расписание было сделано основным |
| till | Это поле будет заполнено, когда расписание на день изменяется. У убранного расписания оно хранит дату, когда оно перестало быть основным |
| lesson | Объект типа LESSON для урока |

**REPLACE**

Коллекция объектов, содержащих информацию о заменах назначенных заменах уроков в классе. Атрибуты:

| Название | Описание |
| --- | --- |
| date | Дата заменяемого урока |
| order | Порядковый в пределах дня номер заменяемого урока |
| venue\_id | ID нового кабинета, если указан |
| teacher\_id | ID заменяющего учителя, если указан. Если нет, то должен быть ID предмета. |
| curriculum\_id | ID предмета, которым заменяют, если указан. Если нет, то должен быть указан ID заменяющего учителя |

**ATTENDANCE**

Объект, описывающие отсутствие человека на уроке. Атрибуты:

| Название | Описание |
| --- | --- |
| date | Дата, когда было замечено отсутствие |
| lesson\_order | Порядковый в дне номер урока, на котором было замечено отсутствие |
| student\_id | ID отсутствующего ученика |

**LESSON**

Содержит в себе данные об одном из уроков из расписания. Атрибуты:

| Название | Описание |
| --- | --- |
| order | Цифра, отражающая номер урока согласно порядку в расписании |
| curriculum\_id | ID предмета урока в базе. Берется из объекта типа CURRICULUM |
| venue\_id | ID кабинета урока в базе. Берется из объекта типа VENUE |

**CURRICULUM**

Объект, отображающий один из предметов в школе. Его ID часто используются в других объектах, например в LESSON выше. Атрибуты:

| Название | Описание |
| --- | --- |
| name | Внутреннее для школы обозначение предмета |
| alias | Отображаемое в системе название предмета |
| master\_id | ID главного преподавателя предмета |
| student\_ids | Необязательный атрибут, содержащий в себе ID учеников. Применяется для кружков и уроков с разделением классов, чтобы хранить информацию о том какой ученик какие конкретно предметы посещает |

**LESSONTIME**

Хранит информацию о временах прохождения уроков в течение дня. Атрибуты:

| Название | Описание |
| --- | --- |
| name | Внутреннее школьное обозначение для данного сочетания времен |
| time | Коллекция объектов TIME для расписания |

**TIME**

Объект с информацией о времени проведения урока в течение дня. Атрибуты:

| Название | Описание |
| --- | --- |
| order | Порядковый номер в расписании расписания для конкретного времени урока |
| from | С какого времени идет урок |
| till | По какое время идет урок |

**VENUE**

Содержит информацию о каждом отдельном кабинете в школе. Атрибуты:

| Название | Описание |
| --- | --- |
| name | Название кабинета для отображения |

**PEOPLE**

Этот объект объединяет информацию о любом человеке в базе относящемуся к школе. Это может быть кто угодно - ученик, учитель, родитель или же наставник. Поэтому и набор данных, предоставляемых этим объектам, бывает крайне разный. Атрибуты:

| Название | Описание |
| --- | --- |
| firstname | Имя, |
| middlename | Отчество, |
| lastname | Фамилия человека. |
| email | Адрес электронной почты, к которой привязан аккаунт |
| birthday | Дата рождения человека |
| type | Один или несколько типов аккаунта, которыми может быть человек |
| student\_ids | Атрибут аккаунтов типа “родитель” - содержит список ID детей человека |
| class\_ids | Атрибут аккаунтов типа “наставник” - список ID классов, для которых человек является наставником |

**NODE**

Этот объект описывает одну из точек для создания карты и навигации по ней. Атрибуты:

| Название | Описание |
| --- | --- |
| name | Системное имя данной точки |
| position | Координаты вида (x, y) для корректного расположения данной точки относительно других |

**CONNECTION**

Напрямую связанный с NODE объект. Содержит информацию о связях каждой отдельной точки с другими. Атрибуты:

| Название | Описание |
| --- | --- |
| node\_id | ID точки к которой привязан этот объект |
| connection\_ids | Список ID точек с которыми связана точка указанная в поле Node\_id |

**CHATS**

Коллекция созданных внутри школы чатов между любыми двумя людьми из школы. Каждый объект содержит атрибуты:

| Название | Описание |
| --- | --- |
| people\_ids | Список из ID людей, между которыми идет переписка. Обычно таковых 2 |
| messages | Коллекция объектов MESSAGE для данного чата |

**MESSAGE**

Этот объект хранит в себе информацию об одном сообщении в чате. Имеет атрибуты:

| Название | Описание |
| --- | --- |
| message | Текст сообщения |
| sent\_by | ID человека, отправившего сообщение |
| timestamp | Точное время отправки сообщения |

**HOMEWORK**

Эта коллекция содержит в себе объекты, представляющие собой домашнее задание, а также результаты их проверки и выполнения. Атрибуты:

| Название | Описание |
| --- | --- |
| text | Текст домашнего задания - его главная часть |
| curriculum\_id | ID предмета, по которому задано дз |
| date | Дата, когда оно было задано |
| student\_id | Необязательный атрибут. Выставляется в том случае если задание было задано лично какому-то человеку |
| teacher\_id | ID учителя, задавшего дз |
| completions | Коллекция объектов типа COMPLETION для дз |

**COMPLETION**

Объект, отражающий отметку о (не)выполнении дз учеником. Может быть изменен как со стороны ученика - то есть будет поставлена или же убрана отметка “выполнено”, та ки со стороны учителя - статус изменится на “проверено”. Атрибуты:

| Название | Описание |
| --- | --- |
| status | Текущий статус выполненности дз |
| completed\_by | ID выполняющего ученика |
| confirmed\_by | ID проверяющего - это может быть учитель или же наставник |

**MARK**

Коллекция об оценках для всех и каждого. Атрибуты элементов:

| Название | Описание |
| --- | --- |
| comment | Необязательный комментарий к оценке |
| teacher\_id | ID учителя, поставившего оценку |
| student\_id | ID ученика, котрому оценка была поставлена |
| mark | Сама оценка |
| lesson\_order | Порядковый номер урока в расписании, на котором была выставлена оценка |
| type | Тип оценки - обычная или контрольная |
| date | Дата выставления оценки |
| curriculum\_id | ID предмета, по которому выставлена оценка |

**TEACHERSRATES**

Коллекция отзывов, оставленных учениками учителям. Такое действие ученик может совершить 1 раз в месяц. Каждый объект содержит атрибуты:

| Название | Описание |
| --- | --- |
| rater\_id | ID ученика, поставившего оценку. |
| teacher\_id | ID учителя, которого оценили |
| ratedate | Дата выставления оценки |
| commentary | Необязательный комментарий к оценке |
| rating | Сама оценка - от 1 до 5 |

Все эти коллекции и объекты тесно связаны между собой внутри системы для лучшего пользовательского опыта и доступа к любой информации в любое время.