

1. Прототипы экранных форм (не менее 7)

1.1. Вход

Форма входа в приложение для авторизации старост с мобильного устройства

ВХОД

Логин

Пароль

ВОЙТИ

1.2. Главное меню

Главное меню приложения для навигации между основными окнами, представленными на рисунке

СТАРОSTATS


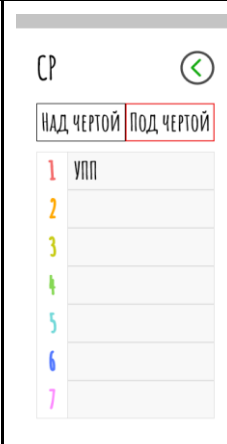

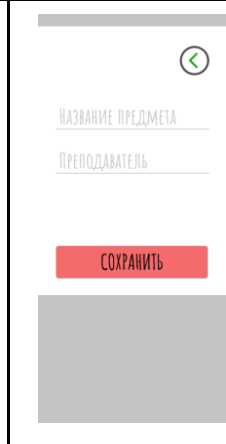

ПОСЕЩАЕМОСТЬ

ОТЧЕТ

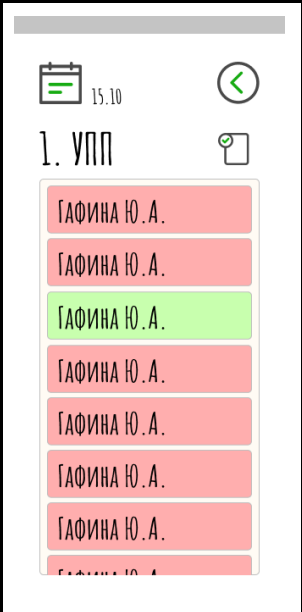
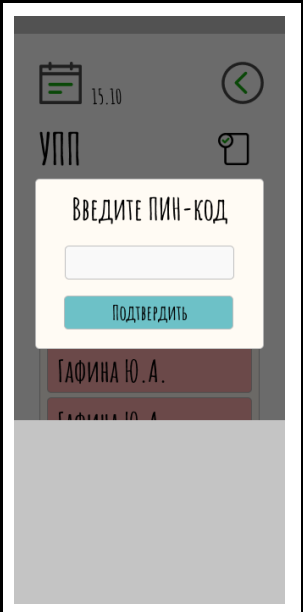

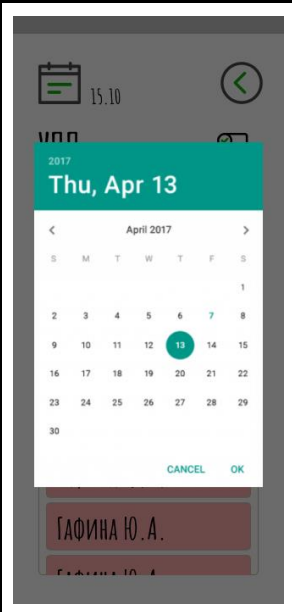
РАСПИСАНИЕ

СПИСОК СТУДЕНТОВ

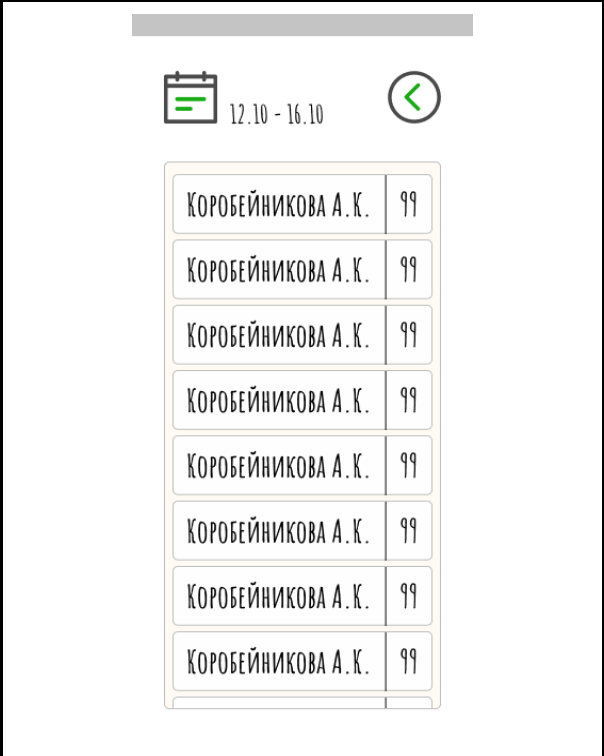
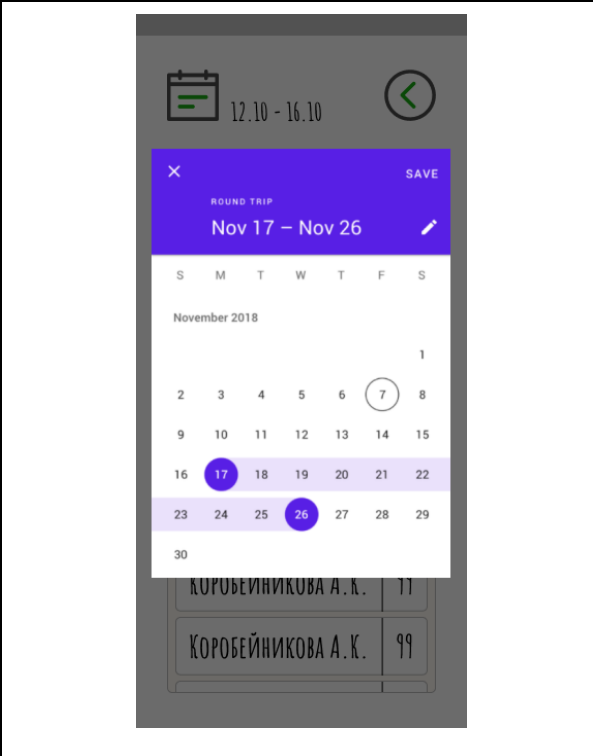
1.3. Составление расписания

				
Выбор дня недели	Добавление предмета в расписание	Выбор предмета из ранее созданных	Создание нового предмета	Выбор преподавателя из списка доступных

1.4. Контроль посещаемости

			
Список студентов для отметки посещаемости	Окно ввода ПИН-кода преподавателем для подтверждения списка присутствующих студентов	Выбор дня и номера пары для просмотра посещения за определенную дату	Выбор даты просмотра списка присутствующих

1.5. Отчет

	
Просмотр списка студентов с количеством пропусков за выбранный период	Выбор периода отчета

1.6. Вход на сайте

ВХОД

ЛОГИН

ВВЕДИТЕ ЛОГИН

ПАРОЛЬ

ВОЙТИ

Вход для сотрудников деканата

1.7. Окно выбора группы и отчет

[illegible]

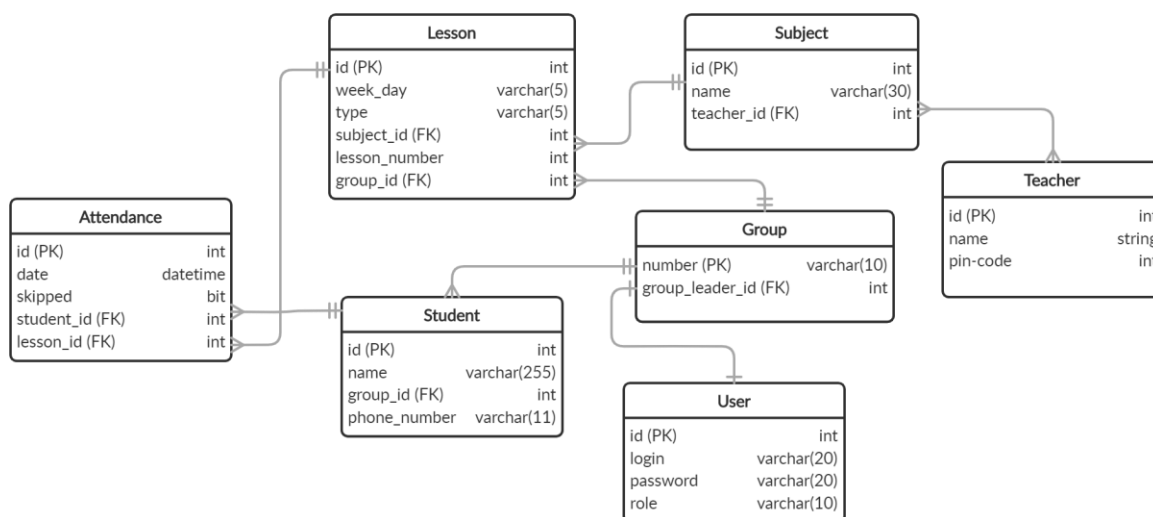
Выбор группы и периода отчета



Б20-191-1	26/10/20 - 31/10/20	
ГАФИНА ЮЛИЯ АНДРЕЕВНА	0	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	10	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	16	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	4	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	0	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	12	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	4	8(000)-000-00-00
ГАФИНА ЮЛИЯ АНДРЕЕВНА	0	8(000)-000-00-00

Отчет посещаемости по группе

2. Диаграмма сущностей (ER) (не менее 7 сущностей)



3. Разработка api системы (не менее 10 функций)

3.1. SignIn(User)

- Выполняет вход пользователя в систему
- Входные данные: объект User
- Выходные данные: токен для авторизации

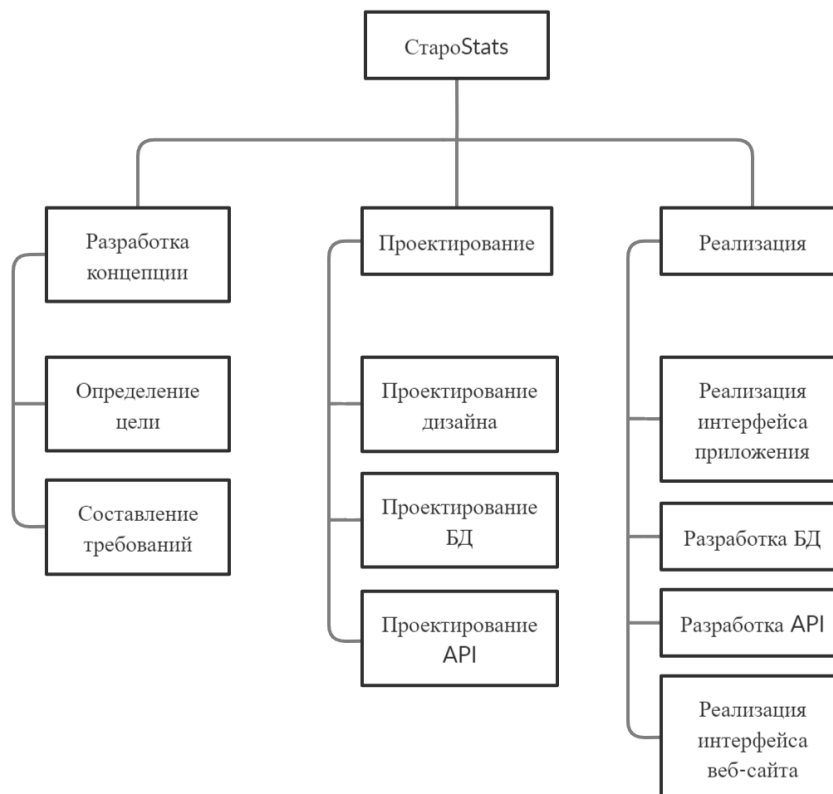
3.2. SignOut()

- Выполняет выход пользователя из системы
- Входные данные отсутствуют

- Выходные данные отсутствуют
- 3.3. AddSubject(Subject)
 - Добавляет новый предмет в базу данных
 - Входные данные: объект Subject
 - Выходные данные отсутствуют
- 3.4. UpdateSubject(Subject)
 - Изменяет информацию о предмете
 - Входные данные: обновленный объект Subject
 - Выходные данные отсутствуют
- 3.5. RemoveSubject(subject_id)
 - Удаляет предмет из базы данных
 - Входные данные: идентификатор предмета
 - Выходные данные отсутствуют
- 3.6. GetSubject(subject_id)
 - Получает предмет из базы данных
 - Входные данные: идентификатор предмета
 - Выходные данные: объект Subject
- 3.7. AddLesson(Lesson)
 - Добавляет новое занятие в расписание
 - Входные данные: объект Lesson
 - Выходные данные отсутствуют
- 3.8. GetLessons(group_id, week_day)
 - Получает расписание группы
 - Входные данные: идентификатор группы, день недели
 - Выходные данные: список объектов Lesson
- 3.9. RemoveLesson(lesson_id)
 - Удаляет занятие из расписания группы
 - Входные данные: идентификатор расписания
 - Выходные данные отсутствуют
- 3.10. GetTeachersList()
 - Получение списка преподавателей
 - Входные данные отсутствуют
 - Выходные данные: список объектов Teacher
- 3.11. GetTeacherPIN(teacher_id)
 - Получение ПИН-кода преподавателя для подтверждения посещаемости
 - Входные данные: идентификатор преподавателя
 - Выходные данные: строка - ПИН-код
- 3.12. GetStudents(group_id)
 - Получение списка студентов определенной группы
 - Входные данные: номер группы
 - Выходные данные: список студентов
- 3.13. GetGroupByLeaderId(group_leader_id)
 - Получение группы по идентификатору старосты

- Входные данные: идентификатор старосты
 - Выходные данные: группа - объект Group
- 3.14. GetGroups()
- Получение списка групп
 - Входные данные отсутствуют
 - Выходные данные: список объектов Group
- 3.15. SetStudentsAttendance(date, lesson_id, students[])
- Добавление пропусков занятий студентами
 - Входные данные: дата, номер пары, id студентов, пропустивших занятие
 - Выходные данные отсутствуют
- 3.16. GetStudentsAttendance(date, lesson_id)
- Получение пропусков занятий студентов за определенный день
 - Входные данные: день, номер пары
 - Выходные данные: (student, attendance)[]
- 3.17. GetReport(group_id, date)
- Получение отчета по посещаемости группы за выбранный период
 - Входные данные: номер группы, временной период
 - Выходные данные: (student, skipped_count)[]

4. Иерархическая структура работ (ИСР)



5. Разбить работы на 2 этапа: MVP и финальная-версия

5.1. MVP:

- Разработка функций по
 - редактированию расписания
 - отметке посещаемости студентов из стандартного списка
- Разработка интерфейса

Все данные хранятся на телефоне пользователя и не отправляются на сервер

5.2. Финальная версия:

- Разработка сайта
- Разработка базы данных
- Разработка полноценного API

6. Оценить время выполнения проекта по методу PERT

Работы	К (шт.)	М (ч.ч.)	О (ч.ч.)	Р (ч.ч.)	Е	СКО
Сущности	7	1	1	2	1,166667	0,166667
Формы	16	3	2	4	3	0,333333
Методы	15	4	3	6	4,166667	0,5
					31,66667	6,293736

- $E_{95\%} = E + 2СКО = 25,66667 + 2 * 5,797509 = 44,25414(\text{ч.ч})$
- $E_{\text{общ}} = E_{95\%} * n = 37,26168 * 3 = 132,7624(\text{ч.ч})$
- $T_p = 165 * 0.8 = 132(\text{ч.ч} / \text{мес.})$
- $T = E_{\text{общ}} / T_p = 111,7851 / 132 = 1,005776(\text{мес.})$

7. Базовое расписание в виде диаграммы Ганта

