



Проектирование "университет"

Выполнил:
Кошелев Александр

Содержание

1. Проектирование макета сервиса	3
1.1. Постановка задачи	3
1.2. Анализ ролевой модели	3
1.3. Макеты UI	4
1.3.1. Административная структура	4
1.3.2. Персоналии	5
1.3.3. Учебный план	5
1.3.4. Журнал	6
2. Проектирование API	6
2.1. Выявление достаточных DTO	6
2.1.1. Административная структура	6

1. Проектирование макета сервиса

1.1. Постановка задачи

Задача в данном проекте ставится достаточно просто – разработать бэкенд учетного сервиса.

Необходимо иметь возможность оперировать как крупными структурами вроде факультета или направления обучения, так и мелкими, например, студентами. Также необходимо проанализировать ролевую систему в рамках приложения, проработать возможный вид UI и строить бэкенд методом "от потребителя".

1.2. Анализ ролевой модели

Поскольку система подразумевает самые разные уровни управления, все слои иерархии института должны получить соответствующие UI и функционал для ведения деятельности. Предполагаются следующие роли и полномочия:

- Ректорат – просмотр и редактирование факультетов и направлений обучения.
- Дирекция факультета – изменения в структурах курсов, предметов, пересмотр учебной части.
- Кадровая служба – прием на работу преподавателей, изменения в кадровой структуре.
- Приемная комиссия – прием на обучение студентов.
- Учебные управления факультетов – пересмотр оценок студентов в исключительных случаях.
- Преподаватели – редактирование оценок студентов.
- Студенты – просмотр оценок и учебного плана.

На основании такой ролевой модели можно выделить необходимые для функционирования сервиса единицы GUI. Представим их в табличном виде по отношению к имеющимся ролям и подробно опишем ниже.

Роль/GUI	Административная структура	Персоналии	Учебный план	Журнал
Ректорат	EDIT	READ	READ	READ
Дирекция факультета	READ	READ	EDIT	READ
Кадровая служба	READ	EDIT Кадровая структура	READ	READ
		READ Остальное		
Приемная комиссия	READ	EDIT Студенты	READ	READ
		READ Остальное		
Учебное управление	READ	READ	READ	EDIT
Преподаватели	READ	READ	READ	EDIT
Студенты	READ	READ	READ	READ

Табл. 1: Права ролей

Начнем с *административной структуры*, которая будет представлять из себя набор инструментов для просмотра и редактирования факультетов и направлений обучения. Далее *персоналии* — раздел с лицами института, преподавателями и студентами. *Учебный план* будет набором инструментов планирования курсов, расписания и предметов с учетом распределения нагрузки и занятости преподавателей и студентов. Наконец, *журнал* — система учета успеваемости обучающихся.

1.3. Макеты UI

Прикинем примерный интерфейс соответствующих инструментов.

1.3.1. Административная структура

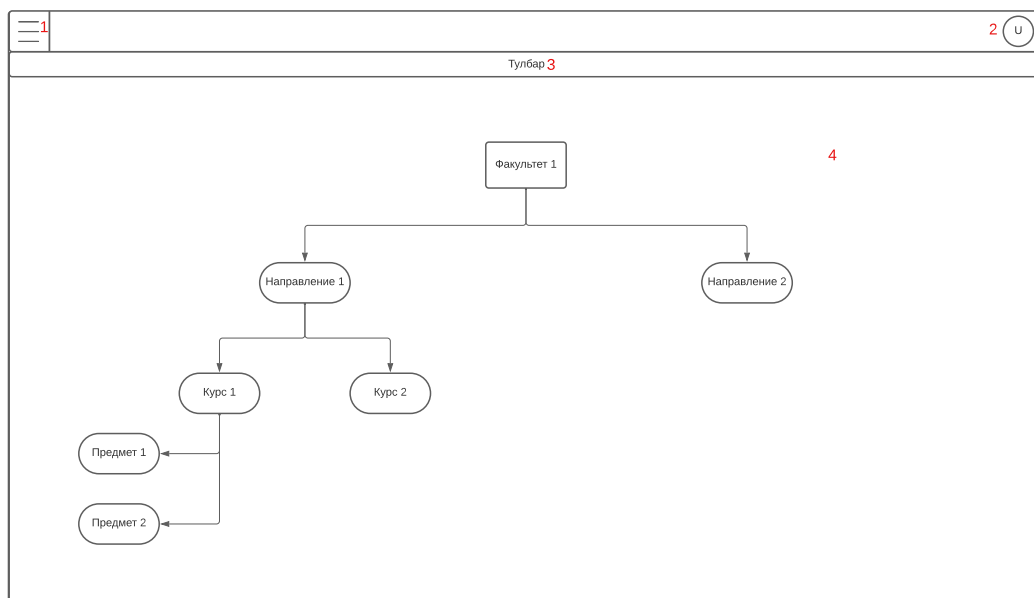


Рис. 1: UI административная структура

- 1) Navigation drawer – навигация по сайту (не требует бек)
- 2) ImageButton – переход в личный кабинет (требуется информация о пользователе)
- 3) Toolbar – инструменты фильтрации и настройки глубины графа (требуется фильтры на сущности)
- 4) Graph – рабочее поле с возможностью перехода на страницы соответствующих звеньев (требуется данные о сущностях)

1.3.2. Персоналии

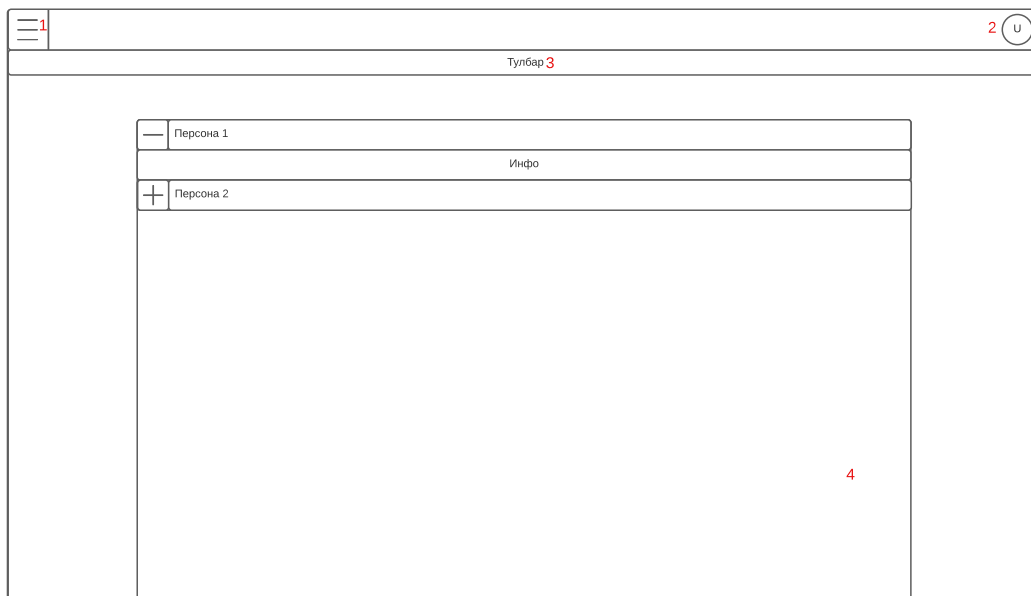


Рис. 2: UI персоналии

- 1) Navigation drawer – навигация по сайту (не требует бек)
- 2) ImageButton – переход в личный кабинет (требуется информация о пользователе)
- 3) Toolbar – инструменты фильтрации и выборки (требуется фильтры на сущности)
- 4) List – рабочее поле с возможностью перехода на страницы соответствующих персон (требуется данные о сущностях)

1.3.3. Учебный план

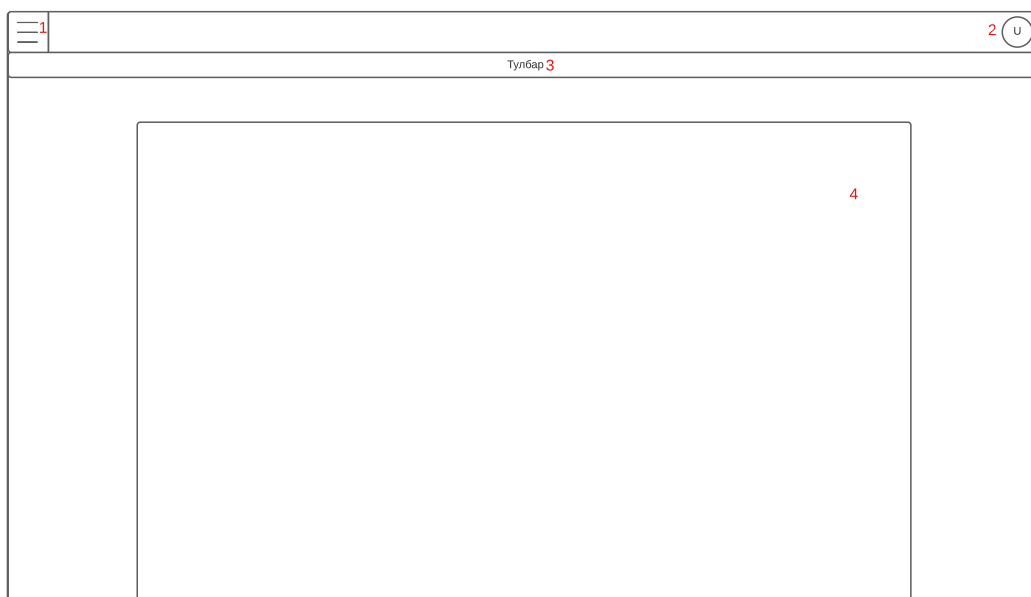


Рис. 3: UI учебный план

- 1) Navigation drawer – навигация по сайту (не требует бек)
- 2) ImageButton – переход в личный кабинет (требуется информация о пользователе)
- 3) Toolbar – инструменты фильтрации и выборки (требуется фильтры на сущности)

- 4) Table – рабочее поле, сводная таблица учебного плана (требуется данные о сущностях)

1.3.4. Журнал

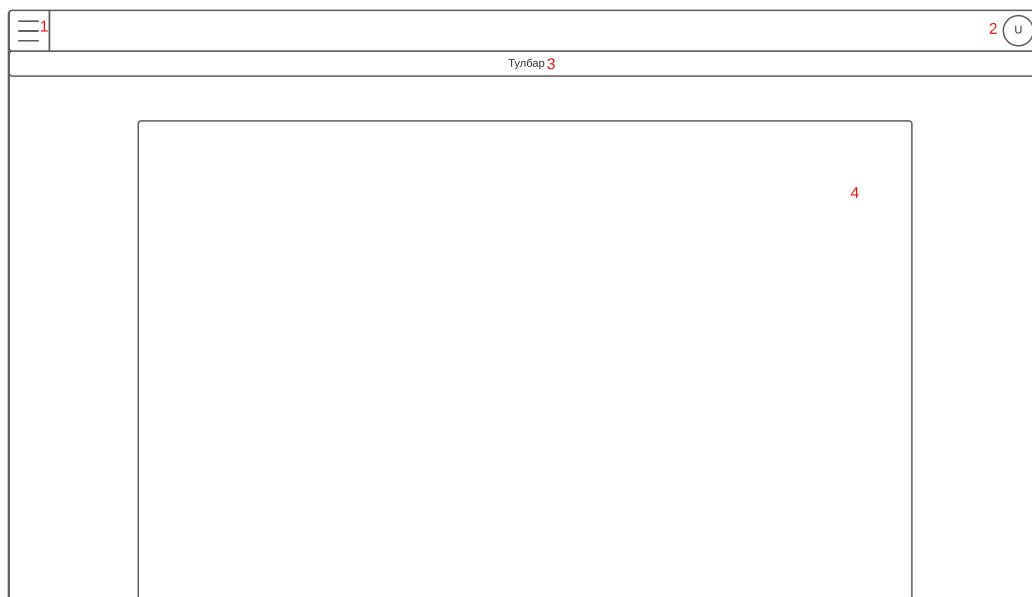


Рис. 4: UI журнал

- 1) Navigation drawer – навигация по сайту (не требует бек)
- 2) ImageButton – переход в личный кабинет (требуется информация о пользователе)
- 3) Toolbar – инструменты фильтрации и выборки (требуется фильтры на сущности)
- 4) Table – рабочее поле, таблица успеваемости студентов в группе (требуется данные о сущностях)

2. Проектирование API

2.1. Выявление достаточных DTO

Рассмотрим, какая информация необходима нашему UI для его работы, и отобразим, какие поля должны быть у наших DTO, чтобы эту информацию доставить.

2.1.1. Административная структура

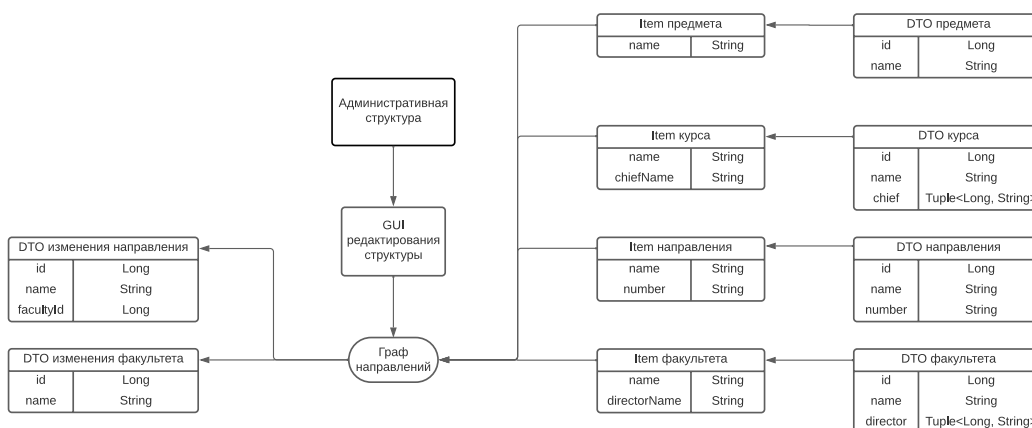


Рис. 5: DTO административная структура