

FLATMEET

Обедини съседите си в една общност!

1. Цели:

Основната цел на приложението е да обединява съседите от даден вход в общност, докато се запазва комуникацията във времена на социална дистанция и самоизолация. Приложението може да се използва и след кризата – примери:

Довечера ще правя барбекю пред блока. Заповядайте!

Зелето трябва да се протака, докато съм на ски. Има ли желаещ да помогне?

Навити да боядисаме пейката пред блока този уикенд? (анкета)

Друга важна цел е да се улесни събирането на паричните средства, с които разполагат най-често така наречените касиери във всеки вход (примери: събиране на пари за асансьор, чистач, обща вода и т.н.), като се онагледява интерактивно процесът чрез таблици и графики, до които всички съседи имат достъп.

2. Функционалност

а. Обща

1. Създаване на вход

1. Всеки потребител може да създаде вход, в който той автоматично става администратор

2. При създаването се изисква пълният адрес на входа

2. Когато даден потребител за пръв път влезе във вход, системата изисква да напише ет., ап. евентуално семейство

3. Смяна на име, парола

4. Напускане на вход

5. Членуване в повече от един вход (един и същи потребител може да има различен ранг в различните входове).

б. Социална

і. Комуникация под формата на текстов чат с останалите съседи

с. Счетоводна

1. Добавяне на неограничен брой общи разходи на входа.

2. За всеки общ разход ще предлагаме различни начини за разделяне между живущите:

1. Поравно между апартаментите
 2. Поравно между всички живущи
 3. Ръчно въвеждане на процент от общата сума
 4. Допълнително, даден апартамент ще може да бъде изключван от сметките по даден разход (като асансьорни такси).
3. Показване по месеци, апартаменти и различни разходи кой апартамент колко е дължил и дали е платил.

Приложението е трислойно.

1. Най-външният слой е преди влизането или регистрацията на потребителя. Функционалности:
 1. Логин
 2. Регистрация
 3. Забравена парола
2. Вторият слой е след влизането в системата, но преди да влезе в определена общност (всеки потребител може да членува в безброй много общности, тъй като може да живее на няколко места. Пример – студенти). Функционалности:
 1. Показване на всички общности, в които членува потребителят
 2. Присъединяване към общност
 3. Създаване на общност
3. Най-вътрешният слой е при присъединяването към дадена общност. Потребителите биват 3 вида:
 1. Съсед
 2. Касиер
 1. Само той има достъп до точки 1 и 2 от счетоводната функционалност.
 2. Повишава съсед до касиер
 3. Администратор (домоуправител)
 1. Има право да отстранява други потребители

2. Одобрява потребителите при влизане
3. Повишава даден потребител до касиер или администратор
4. Понижава касиер до потребител

3. Реализация (обосновка за използване технологични средства, алгоритми, литература, програмни приложения и др.):

1. jQuery – една от най-популярните JavaScript библиотеки с удобен и лесен синтаксис; улеснява ползването на AJAX заявки към back-end; представя данните във приличен JSON формат
2. Bootstrap – популярна и широкообхватна CSS библиотека
3. Laravel 8 – много мощен и пълен с възможности фреймуърк с елегантен и експресивен синтаксис; чрез Eloquent, който ползва техниката ORM (object-relation mapping), се улеснява значително събирането и представянето на данните от базата данни, като се използват практиките на обектно ориентираните езици като PHP; middleware и guards ограничават достъпа на заявки от неустоверени потребители; така наречените broadcasting channels позволяват обмяната на информация в реално време чрез WebSocket технологиите
4. Laravel Echo – JavaScript библиотека, която улеснява процеса на "слушане" за обмен на информация през някой от излъчващите канали
5. MySQL база данни – SQL система за управление на база данни с конвенционален синтаксис