# Сайт за онлайн разплащания

Трябва да се реализира сайт за онлайн разплащания на финансова институция. В него потребителите трябва да могат да се логват, да проверяват наличните суми по сметките си, да извършват плащания и да виждат справка с наредените плащания.

## Обектен модел

### сметки

Всяка сметка има:

* уникален идентификатор
* номер на сметка
* налична сума по сметката

### потребители

Всеки потребител има:

* уникален идентификатор
* имена на потребител
* потребителско име (което се използва за логин)
* парола

### релации потребители - сметки

Позволява се една сметка да е абонирана за повече от един потребител. Не може да има сметки, които не са абонирани към нито един потребител.

Един потребител може да е абониран за много сметки, не може да е абониран към една сметка повече от веднъж (нека тази уникалност да се осигури). Може да има потребител без сметки.

За целите на задачата сметките, потребителите и релациите между тях се въвеждат ръчно от разработчика в базата от данни (т.е. не е необходимо да се разработват средства и функционалност за регистриране и управление на сметки и клиенти).

### Плащания

Атрибутите на плащането са:

* Сметка, от която е извършено плащането
* Сметка, към която е извършено плащането
* Сума - число с 2 знака след десетичната запетая
* Основание на превода - низ с максимална дължина от 32 символа

## Функционалности на приложението, които трябва да се разработят

### Логин на потребители

Вход в приложението, след успешна проверка на потребителско име/парола за съответния клиент.

### Показване на списък сметки

Визуализация на основна страница със списък сметки на потребителя с наличната сума по всяка сметка

### Създаване на ново плащане

Форма, в която всички атрибути са задължителни за избор (въвеждане):

* Сметка, от която да се извърши плащането – трябва да се визуализира списък сметки, за които е абониран потребителят с информация „номер – налична сума“ и възможност за избор от този списък
* Сметка, към която да се извърши плащането – поле с валидация точно 22 символа, само цифри и латински букви
* Сума – число с два знака след десетичната запетая
* Основание – низ с максимална дължина от 32 символа
* При създаване плащането се записва в статус ИЗЧАКВА

### Показване на списък плащания

Визуализират се всички плащания, записани от потребителя с техните атрибути и текущ статус, с възможност за сортиране по два различни критерия – хронологично (от най-новите към най-старите) и по статус (тези със статус ИЗЧАКВА най-горе)

### Изпращане на плащане

Операцията е възможно само ако има достатъчна налична сума по сметката, за да се извърши плащането

Плащането трябва да е в статус ИЗЧАКВА

След успешно изпращане статусът му става ОБРАБОТЕН и намалява съответно наличната сума по сметката, от която се извършва плащането

### Отказване на плащане

Плащането трябва да е в статус ИЗЧАКВА

След отказ статуса му става ОТКАЗАН

## Изисквания и насоки за изпълнението

Начинът на изпълнение, интерпретацията на функционалните изисквания и т.н. са изцяло по преценка и разбиране на разработчика.

Софтуерната архитектура, дизайнът на базата от данни и потребителският интерфейс трябва да бъдат дефинирани и реализирани от разработчика с демонстриране на добри практики за софтуерна разработка – 3-слойна архитектура, подходящ обектен модел, дизайн на базата от данни с необходимите constraint-и т.н. Препоръчителната UI технология е MVC.

Задачата е предвидена за изпълнение в прилично работещо състояние на основните функционалности за около 1 пълен работен ден за самостоятелна работа от един разработчик.

Допълнително (извън рамките на изпитното време от 8 часа), след предаване на работата разработчикът трябва да състави и кратък списък за неизпълнените от него функционалности, евентуални подобрения или разширения, които би искал да довърши, но не е успял заради ограниченото време от един работен ден.