

### Задание

1. Создайте таблицу б.д. с именем, хешированным паролем и информацией о банковской карте и сумме на ней.
2. Разработайте фейслет (xhtml) с использованием чистых html-тегов и JSF-компонентов для входа пользователя в систему.
3. Добавьте на страницу составной компонент на основе **<composite:interface>** **<composite:implementation>**.
4. Добавьте управляемый бин CardCotroller содержащий: номер карты, имя и фамилия, срок действия карты, проверочный код, сумму платежа. Область действия - запрос. Один из методов бина должен выполнять платеж – вычитывать указанную сумму с карты пользователя.
5. Разработайте еще один фейслет (xhtml) для выполнения платежа в случае входа в систему. Компоненты интерфейса будут связаны по свойствам с управляемым бином.
  - a. Эта и все остальные страницы стройте на основе шаблона с тремя (или более) частями header, footer и content. Header, footer – постоянны, меняется только средняя часть content. Используйте **<ui:composition>** **<ui:define>** **<ui:insert>**.
  - b. На фейслете должна быть команда (кнопка) выполнения платежа с навигацией на страницу результата (успешное подтверждение или отказ транзакции). Для навигации используйте атрибут **action**.
  - c. На странице должна быть ссылка на первую страницу входа (используйте явную навигацию или правила навигации определяете в **faces-config** через **<navigation-rule>**).
  - d. Добавьте валидаторы для проверки значения компонентов пользовательского интерфейса до того, как они будут обработаны бином. Например, имя - не менее двух символов, номер карты - от 8 до 10 цифр и т.п.. Используйте **<f:validateLength>** **<f:validateLongRange>** и др.
  - a. Создайте пользовательский валидатор (класс реализующий интерфейс javax.faces.validator.Validator) для срока действия карты (можно на основе регулярных выражений) и примените его к компоненту пользовательского интерфейса на фейслете. В случае неверного формата генерируйте сообщение FacesMessage и добавляйте его контекст.
  - b. Фейслет должен содержать сообщения об ошибке (выводится рядом с компонентом) в случае оплаты большой суммы с разной степенью важности: INFO, WARN или ERROR. Используйте **<h:message>** и сообщение из бина.
  - c. Добавьте пользовательский конвертер (класс реализующий javax.faces.convert.Converter) для компонента «сумма платежа» конвертирующий сумму в другую валюту (например евро).

- d. Добавьте на страницу логотип (картинку) используя неявные объекты JSF (через ресурсы).
  - e. Создайте составной компонент на странице. Продемонстрируйте его использование.
6. При сборе проекта используйте Maven

### Вопросы:

1. Нарисуйте и объясните архитектуру web приложения на основе JSF
2. Как JSF может поддерживать несколько файлов конфигурации?
3. Как задать минимальную конфигурацию для JSF приложения?
4. Что такое navigation rule в JSF?
5. Может ли отсутствовать Faces-config.xml в JSF?
6. Что такое FacesServlet? Опишите его функции и преимущества.
7. Что такое FacesContext? Каково его назначение? Как получить доступ?
8. Зачем и как используются преобразователи и валидаторы?
9. Опишите порядок создания и использования пользовательского валидатора?
10. Какие теги валидации существуют в JSF?
11. Опишите порядок создания и использования пользовательского преобразователя?
12. Какие теги JSF используются для конвертации?
13. Как отобразить сообщение об ошибках в JSF?
14. Что такое JavaBeans?
15. Схематично напишите код для managed JSF beans.
16. Как и зачем в JSF используют managed JSF beans?
17. Какие области действия может иметь managed JSF beans (дайте пояснения по каждому из них)?
18. Что такое аннотации функций обратного вызова managed JSF beans?
19. На что указывает аннотация @ApplicationScoped?
20. Перечислите неявные объекты JSF?
21. Опишите все способы навигации по страницам в JSF?
22. Что такое Facelets? Что они содержат? Назовите Facelets теги
23. Как написать страницу с использованием тегов шаблонов JSF?
24. Как создать составной компонент на странице?
25. Что такое событие (event)? Какие типы событий существуют в JSF?