02267 - Software Development of Web Services

Development of Mobile Payment System
Project Description

Kolkata

May 1, 2018

Danmarks Tekniske Universitet

Project Description

Architecture

Basic system overview

This is the conceptual drawing of the system overview as provided by Werner Wild during lecture.

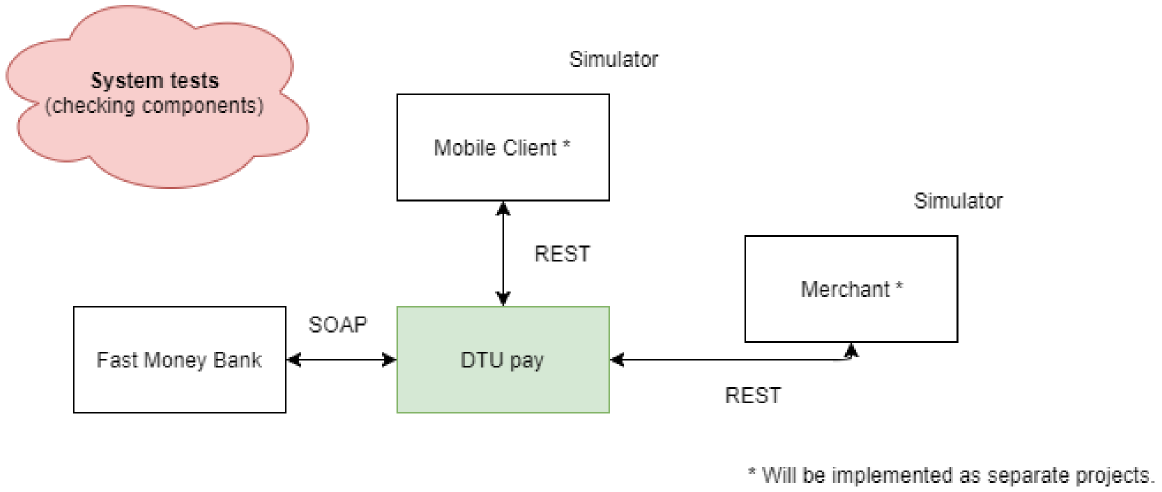

Figure 1: Initial system architecture

DTU pay system overview

This is the architecture for group Kolkata's implementation of the system.

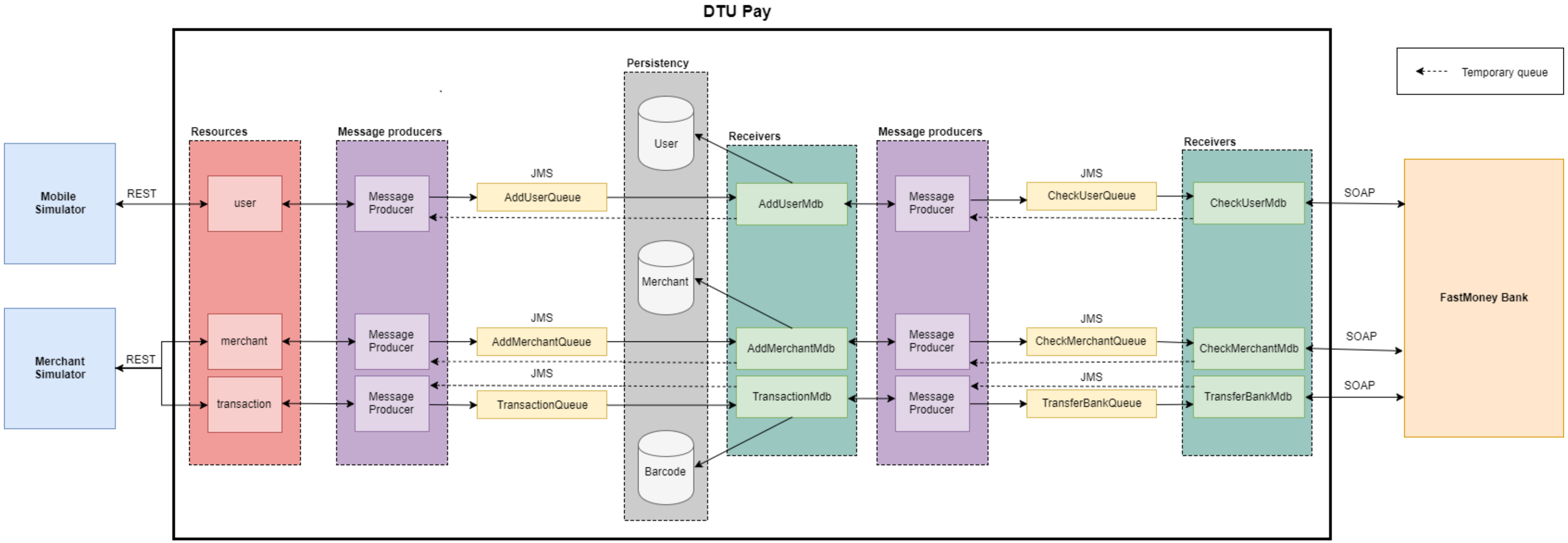

Figure 2: Architecture for the entire system

Message Driven Beans

- **MessageReceiver** (superclass)
  This class is the super class for all Message Driven Beans. It ensures that the onMessage method and parseMessage method are inherited from its superclass. Since this class is abstract there are no objects created.

- **AddMerchantMdb**
  Input message:

  - Type: TextMessage
  - Content: "addMerchant/" + cvr + "/" + name

  Respond type: String

  - Bank Error: "Bank account does not exist with this CVR:" + cvr
  - JMS Error: "JMS Error";
  - Dublicate Merchant: "Merchant already exists."

  - Success: ""New merchant created."

- **CheckMerchantMdb**
  Input message:

  - Type: TextMessage
  - Content: ""checkAccount/" + cvr

  Respond type: String

  - Bank Error: "Account does not exist"
  - Web Service Error: "Web Service Error";
  - Success: "Account exists"

- **AddUserMdb**
  Input message:

  - Type: TextMessage
  - Content: ""addUser/" + cpr + "/" + firstName + "/" + lastName

  Respond type: String

  - Bank Error: "Bank account does not exist with this CPR: " + cpr
  - JMS Error: "JMS Error";
  - Dublicate User: "User already exists."
  - Success: "New user created"

- **CheckUserMdb**

  Input message:

  - Type: TextMessage
  - Content: ""checkAccount/" + cpr

  Respond type: String

  - Bank Error: "Account does not exist"
  - Web Service Error: "Web Service Error";
  - Success: "Account exists"

- **GetBarcodeMdb**

  Input message:

  - Type: TextMessage
  - Content: "getBarcode/" + cpr

  Respond type: String

  - "Storage" Error: "Couldn't find user with the given CPR: " + cpr
  - Success: barcode

- **ResetStorageMdb**

  Input message:

  - Type: TextMessage
  - Content: "resetStorage/"

  Respond type: String

  - Success: "Storage cleared."

- **TransactionMdb**

  Input message:

  - Type: TextMessage
  - Content: "transaction/" + amount + "/" + merchantCvr + "/" + UUID

  Respond type: String

  - Bank Service Error: "Web Service Error"
  - JMS Error: "JMS Error";
  - "Storage Error: "Invalid barcode!"
  - Success: "Transaction successful"

- **TransferBankMdb**

    - Type: TextMessage
    - Content: "transfer/" + amount + "/" + cvr + "/" + barcode

  Respond type: String

    - Bank Service Error: "Web Service Error"
    - Success: "Transaction successful"

RESTful webservices

Overview

All of our REST services are called from a base URL: *"http://159.89.31.4:8080/Kolkata/rest"*

/merchant	POST	Creates a new merchant if the merchant does not already exist
/transaction/transfer	POST	Transfer money from a user to a company
/user	POST	Creates a user if the user does not already exist
/user/barcode/{cpr}	GET	Returns a new barcode to a speific user by taking the cpr of that user