

مستند پروژه

بانکداری + تسهیلات

تهیه کننده: علی غلامی
(آکادمی توسن)

نسخه ۱

زمستان ۱۴۰۱

شرح پروژه

نرم افزار بانکداری مورد نیاز است که قابلیت تخصیص تسهیلات های مالی به مشتریان بانک باشد. این نرم افزار در اختیار کارمندان بانک قرار خواهد گرفت تا بتوانند علاوه بر مدیریت مشتری و حساب ها، به آن ها تسهیلات (وام) بدهند.

نرم افزار های بانکداری بسیار وسیع و جامع هستند و قالباً از اصول حسابداری زیادی پیروی می کنند اما در این پروژه به جز روش محاسبه اقساط به اصل حسابداری خاصی نیاز نخواهد بود. همچنین قواعد تسهیلات در بانکداری اسلامی پیچیدگی زیادی دارد که نیازی به رعایت آن ها در این پروژه نمی باشد.

سناریوی کلی سیستم مورد نظر به این شکل است که کاربر بانک باید بتواند در نرم افزار مشتری تعریف کند و برای آن ها حساب افتتاح کند، سپس ینا به درخواست مشتری پرونده تسهیلات تشکیل دهد و مبلغ وام را به حساب مشتری واریز نماید. سپس طبق جدول اقساط ایجاد شده ماهانه از مشتری اقساط دریافت کنید تا کل وام تسویه شود.

علاوه بر سناریوی اصلی سیستم باید قابلیت هایی نظیر گزارش گیری جهت مدیریت سرمایه بانک نیز در اختیار مدیر بانک قرار دهد. پس سیستم شامل دو نوع کاربر است:

۱- کاربر (کارمند بانک) که عملیات مورد نظر مشتری اعم از افتتاح حساب و تشکیل پرونده تسهیلات را انجام می دهد.

۲- مدیر (رئیس شعبه) که عملیات های مدیریتی نظیر گزارش گیری یا تخصیص بودجه جهت اعطا تسهیلات به مشتریان را انجام می دهد.

نیازمندی های کارکردی

در بخش قابلیت های مورد نیاز نرم افزار شرح داده شده است.

نیازمندی های کاربر (کارمند بانک)

- ۱- کاربر باید بتواند با نام کاربری و رمز عبور خود وارد سیستم شود (لاگین)
 - ۲- کاربر باید بتواند در سیستم مشتری ایجاد کند.
- اطلاعات مشتری شامل نام و نام خانوادگی و آدرس است
 - سیستم باید به هر مشتری یک شماره مشتری (آیدی) منحصر به فرد (غیر تکراری) اختصاص دهد.

- ۳- کاربر باید بتواند لیست مشتریان را مشاهده کند.
- ۴- کاربر باید بتواند مشتریان را با استفاده از شماره مشتری جستجو کند.
- ۵- کاربر باید بتواند اطلاعات هر مشتری را ویرایش کند.
- ۶- کاربر باید بتواند مشتری را حذف کند.
- ۷- کاربر باید بتواند برای هر مشتری یک یا چند حساب افتتاح کند.
 - هر حساب شامل دو مشخصه اصلی موجودی و نوع ارز (ریال یا دلار) است
 - سیستم باید به هر حساب یک شماره حساب منحصر به فرد اختصاص دهد.
- ۸- کاربر باید بتواند به حساب مشتری پول واریز کند (نقدی)
- ۹- کاربر باید بتواند برای مشتری پرونده تسهیلات تشکیل دهد.
 - کاربر باید بتواند مبلغ و زمان بازپرداخت درخواستی مشتری را در سیستم وارد کند سپس سیستم در صورت وجود مبلغ درخواستی در حساب شعبه، پرونده را تشکیل می دهد. در غیر این صورت پیام خطا می دهد.
- ۱۰- کاربر باید بتواند جدول اقساط هر تسهیلات را مشاهده کند.
 - جدول اقساط جدولی است که هر سطر آن یک قسط است و شامل ستون های زیر است:
 - ۱- تاریخ سررسید
 - ۲- مبلغ اصل
 - ۳- مبلغ سود
 - ۴- مبلغ قابل پرداخت (جمع اصل و سود)
 - ۵- وضعیت (پرداخت شده – پرداخت نشده)
 - نحوه محاسبه جدول اقساط باید به روش Amortization انجام شود.
- ۱۱- کاربر باید بتواند مبلغ تسهیلات را به حساب مشتری واریز کند (از حساب بانک به حساب مشتری)
- ۱۲- کاربر باید بتواند از مشتری قسط دریافت کند.
 - بازپرداخت مشتری ممکن است شامل مبلغ یک یا چند قسط باشد.
 - سیستم مبلغ بازپرداخت را از حساب مشتری کسر کرده و به حساب بانک اضافه می کند.
- ۱۳- کاربر باید بتواند آخرین وضعیت جدول اقساط هر تسهیلات را مشاهده کند.

نیازمندی های مدیر (رییس بانک)

- ۱- مدیر باید بتواند با استفاده از نام کاربری و رمز عبور وارد سیستم شود.
- ۲- مدیر باید بتواند موجودی حساب بانک را مشاهده و آن را کم یا زیاد کند.
- ۳- مدیر باید بتواند نرخ سود تسهیلات را تعیین کند.
- ۴- مدیر باید بتواند لیست ۵ تراکنش اخیر شعبه را به ترتیب تاریخ از جدیدترین به قدیمی ترین ببیند.
 - هر گونه تغییر در موجودی حساب ها (بانک و مشتری) یک تراکنش محسوب میشود (پرداخت مبلغ تسهیلات به مشتری، دریافت قسط از مشتری، واریز به حساب و ...)
- ۵- مدیر باید بتواند لیست کلیه تراکنش های شعبه را در قالب یک فایل خروجی بگیرد
- ۶- مدیر باید بتواند گزارش کل سود بانک را ببیند.

نیازمندی های غیرکارکردی

در این بخش ویژگی های کیفی و قیود سیستم توضیح داده شده است.

- ۱- کلیه دیتاهای سیستم باید در یک دیتابیس RDBMS ذخیره شود.
- ۲- دسترسی مدیر و کاربر باید جداگانه باشد. به صورتی که هر کدام به قابلیت های دیگری دسترسی نداشته باشند.
- ۳- ساختار پروژه باید شامل حداقل ۲ ماژول (maven) باشد. به صورتی که ماژول core_banking که شامل قابلیت هایی مثل افتتاح حساب و مدیریت مشتریان است از ماژول loan که مربوط به عملیات تسهیلات است جدا باشد. پیشنهاد می شود برای اجرای عملیات های مربوط به دیتابیس هم از یک ماژول مجزا استفاده شود.
- ۴- استفاده از هرگونه ابزار، فریمورک و library نه تنها منعی ندارد بلکه شامل امتیاز مثبت هم خواهد شد.
- ۵- باید بتوان با استفاده از کامند های maven یک فایل jar قابل اجرا از کل برنامه تولید کرد.
- ۶- عملیات های گزارش گیر (قابلیت های ۵ و ۶ در لیست نیازمندی های مدیر) سنگین خواهند بود لذا باید در background اجرا شوند.
- ۷- انتخاب نحوه پیاده سازی رابط کاربری برنامه (ui) به عهده برنامه نویس است. هم میتوان از CLI استفاده کرد هم از GUI (مانند swing) یا ترکیبی از این دو یا هر ابزار دیگر.

نحوه تحویل پروژه

- ۱- کل پروژه باید روی گیت هاب ارائه شود، به این صورت که فقط یک لینک از ریپازیتوری گیت هاب تحویل گرفته خواهد شد.
- ۲- اسکریپت های sql مورد نیاز جهت ایجاد اسکیمای برنامه و آغاز به کار آن باید روی یک فایل به اسم data.sql در فولدر resources یکی از ماژول های پروژه ذخیره شود و قابل اجرا باشد.
- ۳- مهلت تحویل پروژه ۲۰ روز خواهد بود. یعنی ۲۳ بهمن ماه ۱۴۰۱ آخرین موعد تحویل است.

نحوه ارزیابی و امتیاز دهی

موارد اجباری:

- ۱- قابل build بودن پروژه با یک کامند mvn clean package
- ۲- قابل اجرا بودن فایل data.sql
- ۳- قابل اجرا بودن برنامه با کامند java -jar
- ۴- پیاده سازی کلیه نیازمندی های خواسته شده.
- ۵- استفاده از اصول OOP در پیاده سازی
- ۶- ماژولار، خوانا و تمیز بودن سورس پروژه
- ۷- استفاده از stream api در جاوا ۸

موارد اختیاری (امتیاز مثبت)

- ۱- استفاده از ابزار، فریمورک و کتابخانه های مناسب و گوناگون مثلا ORM ها یا Spring
- ۲- GUI
- ۳- استفاده از GoF Design Patterns
- ۴- پیاده سازی خروجی گزارشی (مورد شماره ۵ از نیازمندی های مدیر) به شکل های مختلف مثل excel یا html.