The Decider

* ***Base Rule***: Base class for all Rules.
  + *Receives an Application and perform some validations over* ***[Application.CBResults].****.*
* ***CbAddSpecialMessagesV1:*** Applies a message to the ***[Application.CBResults].***
  + ***Passed***
* ***CbCreditScoreInvalidV1:*** Validations message codes over SSN for potential frauds. ***[Application.CBResults].***
  + ***PossibleFraud***
  + ***Passed***
* ***CbDeceasedCheckV1:*** Validates Score Class ***[Application.CBResults].***.
  + ***Skipped***
  + ***PossibleFraud***
  + ***Passed***
* ***CbDobCheckV1:*** Validates Year Of Birth ***[Application.CBResults].***.
  + ***PossibleFraud***
  + ***Passed***
* ***CbFraudCheckV1:*** Verifies Fraud Texts in ***[Application.CBResult.Messages]***
  + ***PossibleFraud***
  + ***Passed***
* ***CbReportedNameOnOFACListV1:*** OFAC validation in ***[Application.CBResult]***
  + ***Skipped***
  + ***Failed***
  + ***Passed***
* ***CbRetailerSpecificCreditFloorV1:*** Verifies Application Credit Score required by Retailers. Compares the score against the score required by the Retailer. ***[Application.CBResults].***
  + ***Failed***
  + ***Passed***
* ***CbScoreGlobalFloorV1:*** Verifies the application credit score against a required Global Score. It could be valid but needs to be higher than the Global score. ***[Application.CBResults].***
  + ***Failed***
  + ***Passed***
* ***CbStateSpecificCreditFloorV1:*** Validates application credit score meet minimum credit score required. ***[Application.CBResults].***
  + ***Failed***
  + ***Passed***
* ***RiskViewScoreInvalidV1:*** Verifies regulatory restrictions score = 100 (RegulatoryDecissions), deceased score = 200 (Subject reported as Deceased),insufficientDataScore = 222 (Can’t generate score).
  + ***Dependencies:***
    - *Lexis*
    - *HoradricCube*
* ***SSNAssociatedWithFraudV1:*** Verifies if the SSN already was used by a fraudulent application before. ***[Application.User]***
  + ***Dependencies:***
    - ***Applications***
  + ***PossibleFraud***
  + ***Passed***
* ***SSNEHasExistingContractV1:*** Verifies if SSN is associated with another contract. ***[Application.User]***
  + ***Dependencies:***
    - ***CreditCalcs***
  + ***Passed***
  + ***Failed***
* ***SSNPreviousDeclinedV1:*** Verifies the SSN belongs to another application which was already declined. ***[Application.User]***
  + ***Dependencies:***
    - ***Applications (with same SSN)***
  + ***Failed***
  + ***Passed***
* ***SSNRecentlyAppliedV1:*** Verifies if the SSN belongs to an application already. ***[Application.User]***
  + ***Dependencies:***
    - ***Applications (with same SSN)***
  + ***Failed***
  + ***Passed***
* ***BlackListedStatesV1:*** Verifies if the system can operates with application from a specific state. ***[Application.User]***
  + ***Dependencies:***
    - ***User (That should be part of the context)***
  + ***Failed***
  + ***Passed***
* ***CbInvalidCreditRequiresRiskViewV1:*** Verifies an application has a valid credit.
  + ***Dependencies:***
    - ***Lexis***
    - ***HoradricCube***
  + ***Failed***
  + ***Passed***
* ***CollectFraudPointScoreV1:*** Verifies the application has valid fraudpoints. ***[Application.FraudPoints]***
  + ***Dependencies***
    - ***FraudPoints***
  + ***Blocked***
  + ***Passed***
* ***IPBlackListedV1:*** Verifies if the application was created from an IP contained in the system black list. ***[Application.AppliedFromIpAddress]***
  + ***Dependencies:*** 
    - ***BlacklistIP***
  + ***PossibleFraud***
  + ***Passed***
* ***LowIncomeV1:*** Verifies monthly incomes. ***[Application.MontlyIncome]***
  + ***Failed***
  + ***Passed***

**Dependencies:**

* ***Applications (This should be a micro service)***
* ***BlackListIP (This should be a micro service)***
* ***Lexis (This should be a micro service)***
* ***HoradricCube (This should disappear)***
* ***CreditCalcs (This should be a micro service)***
* ***FraudPoints (This should be a micro service communicating to Fraud Point)***