Project Snap

A bright new future for ITC Data Management

Team: SnapCrcklePop

1

Ashley Allen

Specialist, Invoice to Cash Reporting & Analytics for WK Kellogg Co.

2

What is Invoice to Cash?

- Business activities including – Deduction Management, Invoice Collection, and Cash Application
- Deduction: Customer short pays from retail trade agreements, shortages, fees, etc.

3

The Problem?
Data, Data Everywhere!

4

The Solution…
Project Snap

- One centralized place for all data to be stored and connected via relational database tables
- Changes shall be made in real time and reflected in internal business reporting
- Manage team resources and aid in goal setting
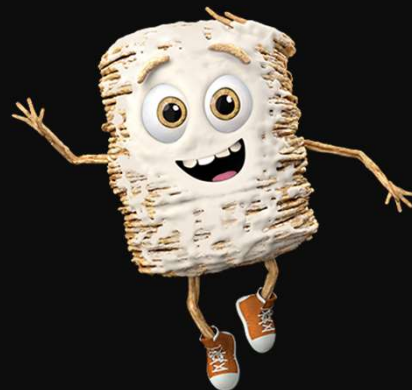- Forecast Deduction clearing based on historical activities

5

Technologies Used

- Python
  - Flask (Backend)
  - Bootstrap (Front End)
- Microsoft SQL Server

In Deployment
- SAP and High Radius
- Deployed on in-house server

6

Features & Requirements
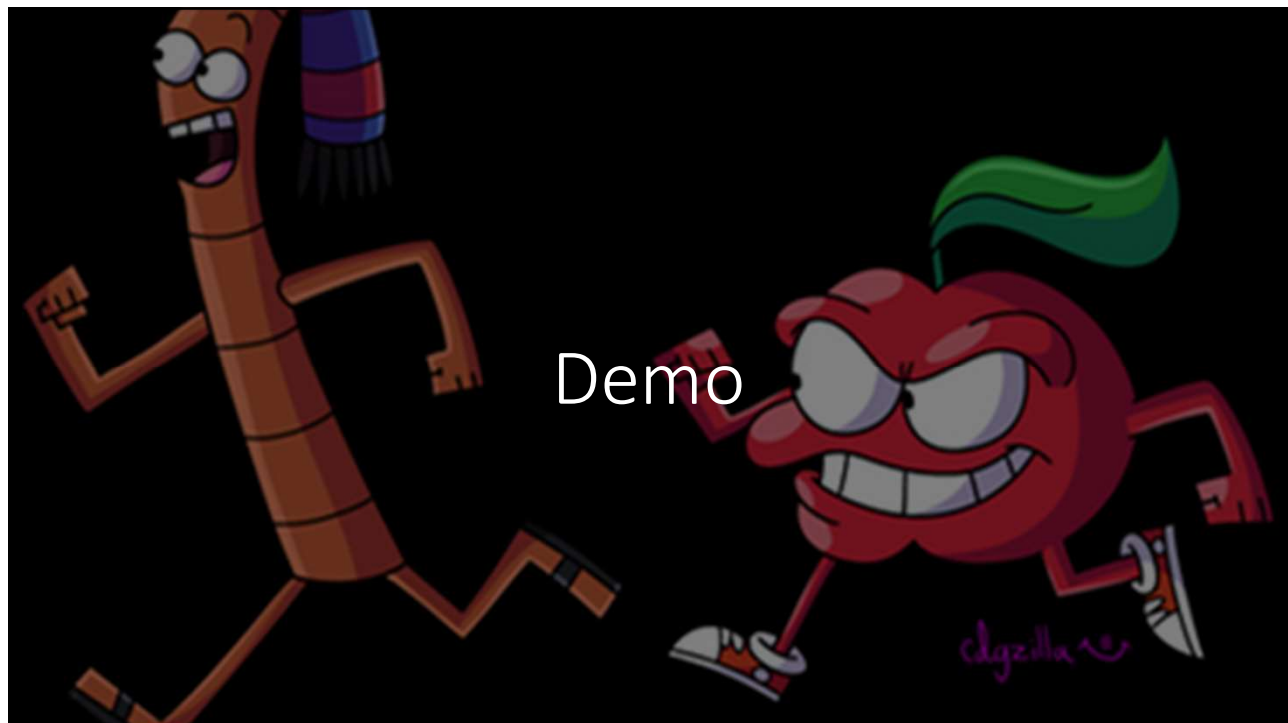
7

Functional Requirements

- User Registration
  - User Friendly Form
  - Required info for registration
  - Password Complexity
  - Confirm Password
  - Display Message
  - Update SQL Table
  - Unique usernames only
- User Authentication
  - Navbar will change based on login status
  - Password Reset Options
  - System Time Out
  - Lock out users
  - Admin to unlock accounts

- Customer Profile and Dashboard
  - Display key metrics
  - Update customer info
  - Only admins can update customer info
  - Outgoing email for support
  - Customer profile picture
- User Profile & Dashboard
  - Users can change their own data
  - Profile picture
  - User data can be visible to all users
  - Deactivate profile
  - Admins to edit data
- Data Entry Operations
  - Data constraints for user data
  - Generate error messages
  - Export data
  - Record user who made change
  - Record time and action of change

8

Non-Functional Features

- Web Based Platform
  - Access across multiple browsers
  - Responsive design
  - Multiple devices
  - Consistent UI design
  - User Friendly Error Messages
- Security
  - Only WK Data
  - Backend data access requires NDA
  - Encrypted Data
  - User account lock-outs
  - Role based access control
- Data Integrity
  - Use SSMS
  - Data ties out with master source
  - Record count tie out
  - No duplicates in data transactions
  - Ensure that data is available and validated daily

- Performance, Usability & Maintainability
  - UI responds quickly
  - Load dashboard within 3 seconds of request
  - Queries return in a max of 2 seconds
  - System messages shall be easily identified
  - Easy to navigate and use
  - Code is fully documented
  - 100 users without degradation
  - UI elements consistent
  - Regular system backups
  - Version control
- Technical Documentation
  - Target audience clearly defined
  - Sections clearly identified
  - Revision history
  - User feedback mechanism
  - Hosted on WK GitHub

9

Demo

10

Slide 11

* resetPassword(userID): null
* updateSupportAccounts(customerID,action): null
* getOpenBalance(accountsSupported): float
* addProfilePic(image): null
* getAtRisk(accountSupported): float

* editUserFirst(userID)
* editUserLast(userID)
* editUserEmail(userID)
* editUserManager(userID)
* editUserRole(userID)
* unlockAccount(userID)

analyst

+ accountsTrained: customer array

- updateTrainedAccounts(customerID,action): null

sales

manager

+ teamRole: int

- getOpenBalance(teamRole): float

director

- strategy: int

- getOpenBalance(): float

0..*   1..1   0..*   0..*

Challenges

No background in python/flask so my design was not relevant to much of the design

OTC_AR

- connectionString: str
- isValid: boolean
- recordCount: int

+ exportData(): null
- tieOut(): boolean
- recordCount(): int
- duplicateCheck(): boolean

- sessionID: int

- validPassword(userID): boolean
- getSessionTime(userID): int
- validSession(userID): boolean
- validNewPassword(userID): boolean
- uniqueAccount(userID): boolean
- uniqueCustomer(customerID): boolean
- activeCustomer(customerID): boolean
- recordChange(before, after, userID, changeTime): null

0..*   0..*

customer

+ customerID: str
+ customerName: str
+ customerNbr: str
+ customerType: int
+ customerChannel: int
+ accountStatus: int
+ accountAge: int
+ supportTeam: user array
+ soldToNbr: str
+ payerNbr: str
+ altPayer: str
+ altPayer2: str

11

Slide 12

Deployment Strategy

• Still a lot of work to do
  • Review Design
  • Code cleanup
  • Security measures
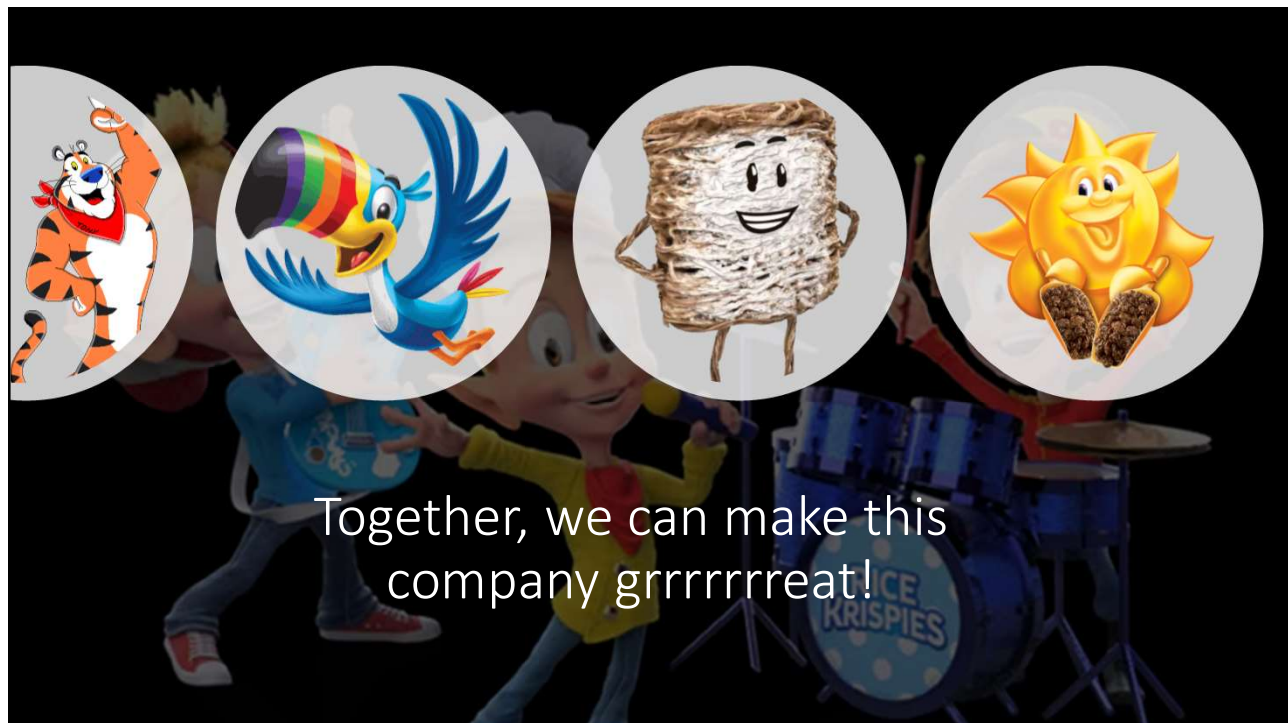 • Work with IT for additional security implementation
  • User testing

12

Training and Implementation Plan

- Live demonstration and Q&A session
- User Testing to ensure the app meets user expectations
- Accuracy Improvement Demonstration in Reporting System
- Recorded Training Video
- Documentation and Support Materials
- Develop a clear implementation timeline and direct all new request to the new system
- Feedback loop

13

Together, we can make this company grrrrrrreat!

14