**Long Beach City Small Business Delinquency**

Project Team Members include the following

Project Advisor: Birgit Penzenstadler

Project Sponsor and Domain Expert: Nishchal Chaudhary

Project Team Leader: Anton Shirokov

Project Developers: Anton Shirokov, David Taitingfong, Andre Barajas, Sovathana Heng and

Ernest Arreola.

**Executive Summary**

**Context**: The City of Long Beach

**Problem**: wants an efficient way to identify delinquent local small businesses. Despite of an insurmountable amount of available data and resources, the department does not have efficient processes established to identify delinquent businesses in need of assistance in a timely manner.  The plight of the app is to implement machine learning techniques and pipelines to extract necessary data and swiftly identify impacted businesses.

**Contribution**: The city government will have an opportunity for a fast response to address the needs of the delinquent small businesses.

**Impact**: maintaining local economy.

**Specification Elaboration**: This document is a preliminary system vision outline of the project details pertaining to the City of Long Beach and its initiative to predict and prevent business delinquency among local businesses based on past and current data and market trends. Using Artificial Intelligence and its machine learning capability, the city of Long Beach will work in partnership with CSULB, to design and develop a solution to this problem.

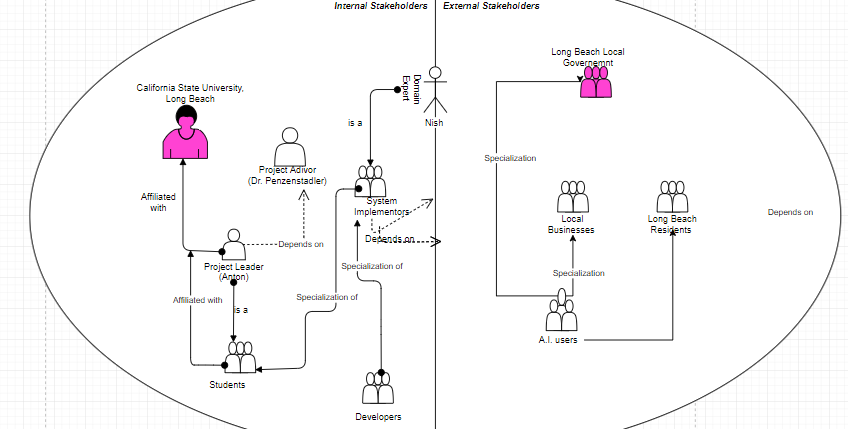
**Stakeholder Model and Description/Relation**

**Stakeholders**:

* City Department represented by Nishchal Chaudhary:
  + **Who:** Local government
  + **Role:** End user,  main funder
  + **Function:** Provide a consultation to the small businesses, interact with the system
  + **Main interest/concern:** Their main interest is to accurately identify delinquent small businesses in a timely manner to take further action.
* Long Beach residents
  + **Who:** Locals that use the money to shop at local small businesses
  + **Role:** Consumer needed for small businesses to thrive, impacted indirectly by the system
  + **Function:** Purchase goods from small businesses
  + **Main interest/concern:** Their main concern is the sustainability of the local enterprises and, therefore, the local community.
* Small Business owners
  + **Who:** Entrepreneur
  + **Role:** owner of the data being collected and analyzed, impacted directly by the system
  + **Function: Information providers**

**Main interest/concern:** get outreach and support from the city if the system determine they are delinquent small businesses.

**Stakeholder Diagram**:



**Goal Model and Description**

**Goals:**

Business: (System Purpose)

1. Identify small businesses in need of help/consultation
2. Reach out to all those small businesses
3. Provide help and consultations
4. Minimize the impact of delinquent businesses on the community
5. Timely detection of delinquent businesses

Usage: (user interacts)

1. Small businesses generate and submit data
2. Local residents provide feedback
3. Establish good relationship between City of Long Beach and small businesses
4. Allow small businesses to submit data directly to the system
5. To improve the city economy as a whole(GDP)

System: (constraints)

1. Analyze small business data
2. Accurately identify delinquent businesses
3. Integrate different data source
4. Manage sensitive data
5. Accessibility control
6. Use data obtained by the City of Long Beach
7. Merge data submitted by small businesses with existing data
8. System maintainability
9. Easy to use interface

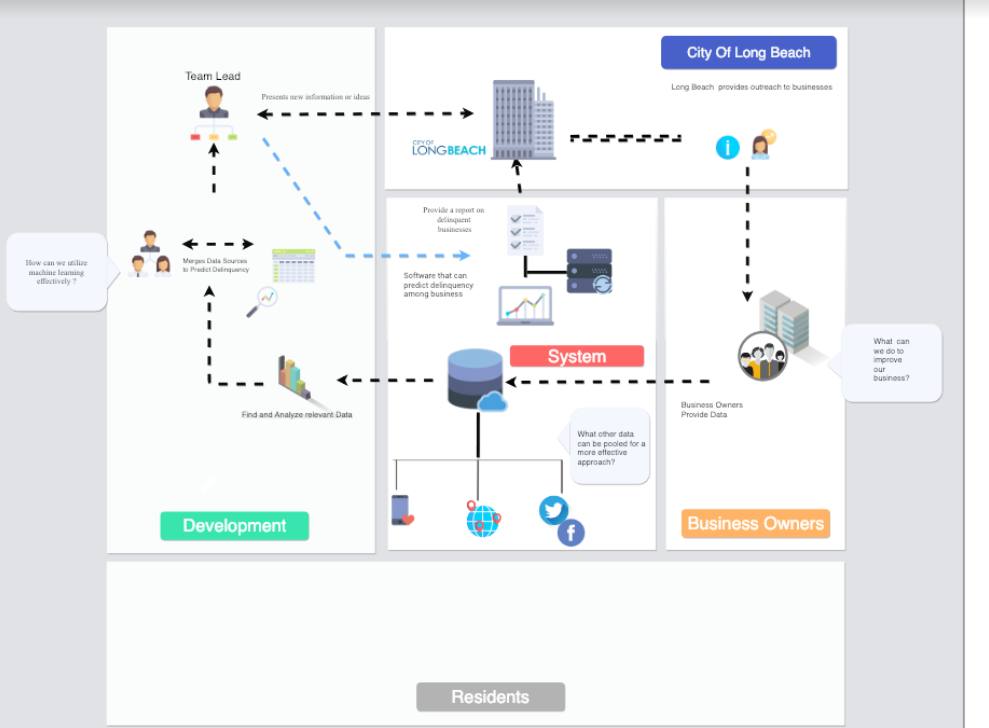
**System Vision and Description**

**Description**: The developers build a system based on the city department’s specifications.

**Vision**: The system is pipelined appropriate data from Long Beach small businesses. This data is analyzed using machine learning techniques.

**Operational and Business Context**: The system identifies delinquent small businesses. The city then takes appropriate action to address presented issues.

**Most Important Stakeholders**: Small local businesses



**Usage Model**

* **Use Case:** To Search for Business
* **CHARACTERISTIC INFORMATION**
* Goal in Context: User locates desired business
* Scope: Long Beach City app
* Level: Primary task
* Preconditions: User has access to the app
* Success End Condition: User finds business
* Failed End Condition: User doesn’t find business
* Primary Actor: User
* Trigger: Business name entered by user
* **MAIN SUCCESS SCENARIO**

1. User locates search field
2. User types in business name
3. App produces search results
4. User select desired business

* **SUB-VARIATION**
* 4. *condition* selection of different option: *action* click Business Not Found
* **SCHEDULE**
* Due Date: Who Knows

**Use Case:** Login

**Characteristic information**

* Goal in context: Access user sensitive data
* Scope: Dashboard
* Level: Subfunction
* Precondition: account was given or created
* Success end condition: Ability to access sensitive data
* Failed end condition: Not able to access sensitive data
* Primary actor: User
* Trigger: navigated to login page

**MAIN SUCCESS SCENARIO**

1. User navigates to login page
2. Clicks on the username field
3. Enter the username
4. Clicks on password field
5. Enter the password
6. Clicks login button

**SUB-VARIATIONS**

* In step 6, login is denied, retry login

**Related Information**

* Priority: Medium priority
* Performance target: user should login within 5 secs of entering the correct password
* Frequency: Often
* Subordinate use cases: User Enters Data, Result Prediction

**Schedule**

* Due Date: March 2019

**Diagram**

**Use Case:** Enter Business Data

**CHARACTERISTIC INFORMATION**

* Goal in Context: User enter business related data manually
* Scope: System application
* Level: Primary task
* Preconditions: Business exist in database
* Success End Condition: Data is manually added to database
* Failed End Condition: Data is not added to database
* Primary Actor: Business owner, user
* Secondary actor: Server side system
* Trigger: User want to add data

**MAIN SUCCESS SCENARIO**

1. User view an existing business
2. User click to add data
3. System provide the categories of data available to be added
4. User enter data
5. Data is added successfully

**SUB VARIATIONS**

* Data type being entered is incorrect
* Business does not exist in the database

**RELATED INFORMATION**

* Priority: Medium
* Frequency: Not too often
* Channel to primary actor: Interactive
* Channel to secondary actor: Internet connection

**SCHEDULE**

* Due date: May 2019

* **Use Case:** View Business Map
* **CHARACTERISTIC INFORMATION**
  + Goal In Context: To view a map (Geo-Location) based system about business in danger of delinquency.
  + Scope:
  + Level: Primary task
  + Preconditions: Has user location.
  + Success End Condition: User sees businesses around.
  + Failed End Condition: User does not see businesses around.
  + Primary Actor: User.
  + Trigger: Opened app.
* **MAIN SUCCESS SCENARIO**

1. User is logged in.

2. App at system level is granted location access.

3. Map is populated with businesses.

4. User can move around and view businesses.

* **EXTENSIONS**

4a. Businesses have indicated markers of delinquency.

* **SUB-VARIATIONS**
  + Can’t connect to server.
* **RELATED INFORMATION** 
  + Priority: High
  + Performance Target: Under 20 seconds to properly propagate.
  + Frequency: Frequent
* **SCHEDULE**
  + TBD 2019

**Use Case:** Check Business Delinquency score

**CHARACTERISTIC INFORMATION**

* Goal in Context: administrator can request a “delinquency” score for business
* Scope: Long Beach City application main portal
* Level: tertiary task
* Preconditions: end user must have admin permissions (security clearance)
* Success End Condition: predictive score is given to a target business
* Failed End Condition: small business has too little information available
* Primary Actor: department
* Trigger:  end user needs a specific business score to determine its need

**MAIN SUCCESS SCENARIO**

1. Application presents a score for a small business based on algorithm

     2. End user utilizes score to compare the need for resources for the business

     3. Score is updated by the application or by admin user

     4. Interface will give a final presentation of possible aspects for improvements for the specific business based on algorithm and data analysis

* **SUB-VARIATION**

·        4. *Add more business information vital to the algorithm predicting more accurate score*

**NFRs**

**Process Requirements**

|  |  |
| --- | --- |
| **Process NFR** | Usability - System should provide visual components to allow better visualization |
| **Rationale** | Allows the user to visualize delinquency rates from week to week |
| **Satisfaction Criterion** | Visualization should be clean and provide minimum abstraction. |
| **Measurement** | Perform survey to 50 people first, more than 70% of the survey should have positive feedback. |
| **Risk** | Visualization will distract users and discourage them from using the system. |

|  |  |
| --- | --- |
| **Process NFR** | Accurate prediction of delinquency score |
| **Rationale** | Allows the user to accurately define small businesses in need of help |
| **Satisfaction Criterion** | The prediction needs to have over 80% accuracy. |
| **Measurement** | Using machine learning models/algorithms, the test data set gave output of over 80% accuracy. |
| **Risk** | City of Long Beach’s delinquency rate of small businesses will not be improved. |

|  |  |
| --- | --- |
| **Process NFR** | Response time should be fast |
| **Rationale** | Good response time will encourage users to use the system |
| **Satisfaction Criterion** | Initial goal: generate a list of delinquency rates should be under an hour |
| **Measurement** | Time the response time when request a new list should be under an hour. |
| **Risk** | Slow response time will disencourage users to use the system |

**Quality Requirements**

|  |  |
| --- | --- |
| **Quality NFR** | Reliability - the system should be accurate |
| **Rationale** | There is no incentive to use the system if isn’t reliable, even semi-reliable. |
| **Satisfaction Criterion** | Heat maps reflect city reports. |
| **Measurement** | Match predictions with past delinquency licenses available from public record. |
| **Risk** | Without accuracy, the app is rendered almost useless. |

|  |  |
| --- | --- |
| **Quality NFR** | Responsive - the system should not be sluggish. |
| **Rationale** | Making an unresponsive interactive map, makes users less likely want to spend more time in the app. |
| **Satisfaction Criterion** | Navigating through app with no hanging. |
| **Measurement** | Compare Web app with other mapping apps to check for similar navigation performance. |
| **Risk** | If the system is unresponsive or hangs excessively, quality will cause user to stop spending time with app. |

|  |  |
| --- | --- |
| **Quality NFR** | System is simple to use. |
| **Rationale** | People with non technical background can use the system. |
| **Satisfaction Criterion** | User can use the system after reading the instructions |
| **Measurement** | Run alpha testing before deployment |
| **Risk** | Misuse of the system lead to wrong delinquent business prediction. |

|  |  |
| --- | --- |
| **Quality NFR** | System is easy to maintain |
| **Rationale** | Can easily update the database when have new data |
| **Satisfaction Criterion** | Person work at the city can add the data into the database |
| **Measurement** | No help needed from Dev team |
| **Risk** | Data is not up to date |

|  |  |
| --- | --- |
| **Quality NFR** | System is secure |
| **Rationale** | Sensitive data is securely stored |
| **Satisfaction Criterion** | No data breach |
| **Measurement** | Pen testing |
| **Risk** | Lose confidentiality of data |

**Deployment Requirements**

|  |  |
| --- | --- |
| **Deployment NFR** | Sufficient data |
| **Rationale** | Data is the key attribute of the system to work as intended |
| **Satisfaction Criterion** | Gather enough data to perform analysis |
| **Measurement** | Perform a sample run before deployment |
| **Risk** | System cannot perform a good analysis |

|  |  |
| --- | --- |
| **Deployment NFR** | Data protection |
| **Rationale** | Protect the private data from exposed to the public |
| **Satisfaction Criterion** | Protect private data is not exposed |
| **Measurement** | Replace private data with dummy data and determine if the data are exposed to the public while migrating |
| **Risk** | Public will get access the private data of the city |

|  |  |
| --- | --- |
| **Deployment NFR** | Easy to integrate |
| **Rationale** | Users should provide minimal effort to integrate the system |
| **Satisfaction Criterion** | Users can follow the instruction and set up the system |
| **Measurement** | Users should be able to set up the system under an hour |
| **Risk** | The help of developers are needed every time the system is being integrated |

**System Constraint**

|  |  |
| --- | --- |
| **System Constraint NFR** | Server accessibility |
| **Rationale** | Should not have time restriction on the access to the system |
| **Satisfaction Criterion** | Server is up 24/7 |
| **Measurement** | Test to make sure the server is up for one week |
| **Risk** | Lost access to server at a random time |

|  |  |
| --- | --- |
| **System Constraint NFR** |  |
| **Rationale** |  |
| **Satisfaction Criterion** |  |
| **Measurement** |  |
| **Risk** |  |

|  |  |
| --- | --- |
| **System Constraint NFR** |  |
| **Rationale** |  |
| **Satisfaction Criterion** |  |
| **Measurement** |  |
| **Risk** |  |

|  |  |
| --- | --- |
| **System Constraint NFR** |  |
| **Rationale** |  |
| **Satisfaction Criterion** |  |
| **Measurement** |  |
| **Risk** |  |

|  |  |
| --- | --- |
| **System Constraint NFR** |  |
| **Rationale** |  |
| **Satisfaction Criterion** |  |
| **Measurement** |  |
| **Risk** |  |