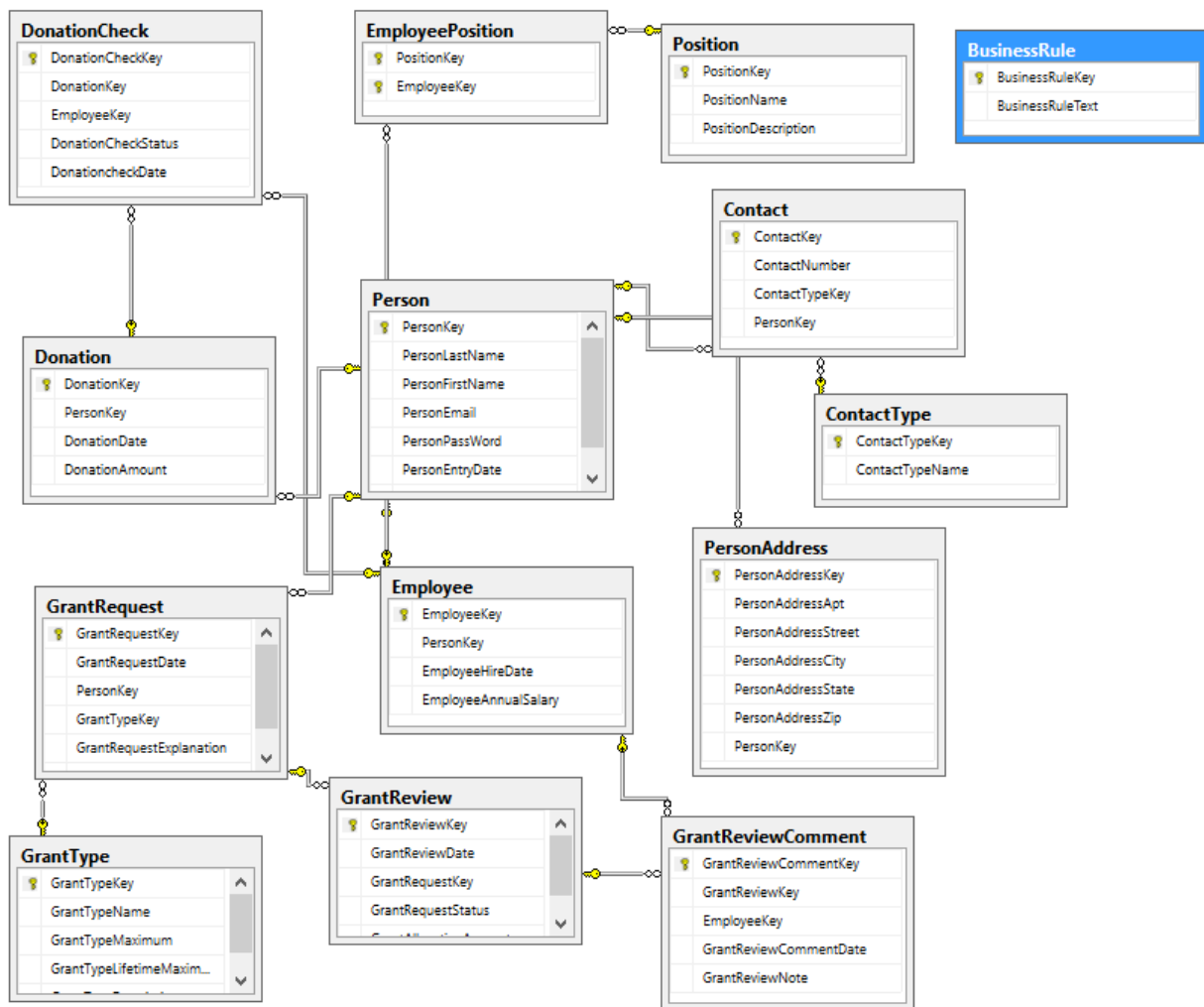# COMMUNITY ASSIST PROJECT

HISTORY

Community Assist is a charity organization that offers one-time loans to community members who find themselves in temporary need. These loans are grouped around predefined services such as Food, Rent, Medical, Education, etc. There are one time and life time limits on the amounts for each of these Services. The goal of the organization is not only to help a client through a hard time, but to help each client find other, more sustainable, means of meeting their needs.

Community Assist is funded through the generosity of donors. Donors can contribute directly to the organization. Each loan is reviewed to insure its authenticity. Donations are not applied to a specific service or client, but are put into a general fund and redistributed as grants. A portion of the donations is used to support the organization its employees. It is the intent of Community Assist that the amount going to charity should always be at least 75% of total donations.

Community Assist is run by a small number of employees. Some are paid and some are volunteer. Employees Assist in data entry, taking donations and processing grants. Each grant request is reviewed by at least one employee. A request for a grant can be approved, approved, but at a lesser amount, or denied.

All activities of the charity are recorded in a database. The database is in a SQL Server Installation which was donated to the charity. The database has been adequate to store the information, but data entry is tedious and difficult and prone to error. This is because there is no adequate application software to serve as a client to the database. Data is entered by hand by the employees. This is time that could better be spent reviewing grant requests, helping clients find sustainable income sources and soliciting donors.

Here is the Entity diagram for the database:

DonationCheck
- DonationCheckKey
- DonationKey
- EmployeeKey
- DonationCheckStatus
- DonationcheckDate

EmployeePosition
- PositionKey
- EmployeeKey

Position
- PositionKey
- PositionName
- PositionDescription

BusinessRule
- BusinessRuleKey
- BusinessRuleText

Contact
- ContactKey
- ContactNumber
- ContactTypeKey
- PersonKey

Person
- PersonKey
- PersonLastName
- PersonFirstName
- PersonEmail
- PersonPassWord
- PersonEntryDate

Donation
- DonationKey
- PersonKey
- DonationDate
- DonationAmount

ContactType
- ContactTypeKey
- ContactTypeName

PersonAddress
- PersonAddressKey
- PersonAddressApt
- PersonAddressStreet
- PersonAddressCity
- PersonAddressState
- PersonAddressZip
- PersonKey

GrantRequest
- GrantRequestKey
- GrantRequestDate
- PersonKey
- GrantTypeKey
- GrantRequestExplanation

Employee
- EmployeeKey
- PersonKey
- EmployeeHireDate
- EmployeeAnnualSalary

GrantType
- GrantTypeKey
- GrantTypeName
- GrantTypeMaximum
- GrantTypeLifetimeMaxim...

GrantReview
- GrantReviewKey
- GrantReviewDate
- GrantRequestKey
- GrantRequestStatus

GrantReviewComment
- GrantReviewCommentKey
- GrantReviewKey
- EmployeeKey
- GrantReviewCommentDate
- GrantReviewNote

THE PROJECT.

The Goal: To build a web based application that can serve the needs of clients, donors and employees.

THE SCOPE:

The project is to build a web based application that serves several distinct needs. For one, it should serve as an informational site for general visitors. A web user should be able to come to the site, see a description of the company including the services offered. They should clearly see how to donate or how to apply for a grant. A general user who wants a deeper understanding should be able to see reports on total donations, total numbers of grants per year and per service, etc. People who wish to donate should be able to register and donate. They also, when logged in as a donor, should be able to see a list of their donations and totals per year. Potential clients should be able to Register and apply for a loan. In addition, they should be able to log in and see the status of their current grant and a summary of any previous grant applications. All database passwords should be hashed and all logins should be logged in the LoginHistory table.

Employees will also access the site, though their access will not be available to the rest of the public. It will be a site to administer loans and donations. Employees should be able to view all applications active and past, select a single application to view. They should be able to comment on and approve, adjust or deny loans. For donations they should be presented with a list of current donations to be confirmed.

Community Assist would like to be able to view the following reports in the application:

- Total Donations for a year grouped by month along with total grants for that year
- Total Grants for a given year grouped by service type
- The total donations, employee costs and grants for a given month/year
- Number of grants per month per year
- Number of grants approved, modified and denied and Percentages for each.

CONSTRAINTS:

The application and database will not store credit card numbers. We will use a Pay pal application to handle credit card transactions for us.

The application will not handle Human Resources activities and processes. This will be dealt with by another second party application.

TIME LINE AND DELIVERABLES

Week One

Overview of Project—scope and requirements

Week Two

Determine possible solutions with feasibility assessment.
Deliverables: Presentation of Feasibility Analysis

Week Three and Four

Determine components of project and requirements assessment (Use Cases)

Deliverables: Use case diagrams and written Use Cases
A test for each use case

Week Five

Analysis of project flow and procedures (Activity Diagrams)

Deliverables: Activity diagrams

Week Six and Seven

Analysis of project structure in terms of services and objects (Class Diagrams)

Deliverable: Class diagrams for each of the domain classes
Relationships among the classes
Code definitions for main classes

Week Eight

Analysis of communication between classes and components (Sequence Diagrams)

Deliverables: Sequence Diagrams

Week Nine and following

Coding and Testing

Deliverables: Code for core functionality. Test Document with Test Results.