MoneyTrail

Financial Money Management and Tracking tool

Contents

Chapter 1

Brief Introduction

This project is an attempt to model the current system of our college IIT Mandi, and to provide tools that enhance the speed and clarity of money flow. In order to achieve this, our goal is to track the money flow in a complex system. From the low level expenditures for food, till the upper level budget allocation from the Government.

After studying some examples, we reached an entity set, that was separated into levels and it had hierarchy among it. It formed a tree, and hence we decided to use names like parents and children. An edge would be interpreted as a money flow from a higher level to a lower level.

Auditing process is also supported here. A special account called Auditor exists that is meant only for viewing and approving "Requests". This is similar to the process of auditing.

Chapter 2

Requirement Analysis

2.1 A Team Member in Design Practicum 2018

I am a student who had taken Design Practicum in this semester. Before the project started, we had to submit proposals, and then request for a budget. It was then transferred to one of my team mates, and we kept track of our expenditures personally. After the project has finished, we had to get the verification done for our bills, and then get the reimbursement for extra money we spent released, after the audit process is completed.

This process is tiresome and takes time. Even though it translates to each team spending a day for the formal procedures, at the larger scale, lot of time is spent in the name of the instutute.

If there was a system that sped up this process, or at least help us in reducing the paperwork, I think that would speed up the process.

2.2 A Club Member

In the beginning of the semester, we propose a budget for the year, and then get it approved by the FA and Dean.

During the semester, we fill a form requesting money for various purposes. before events, some money is also released to us for expenditure. Many times, we spend money from our own pockets, and we preserve the bills so that we can get reimbursed later on.

If there was a system that can make this procedure more transparent and fast, then it would be really nice.
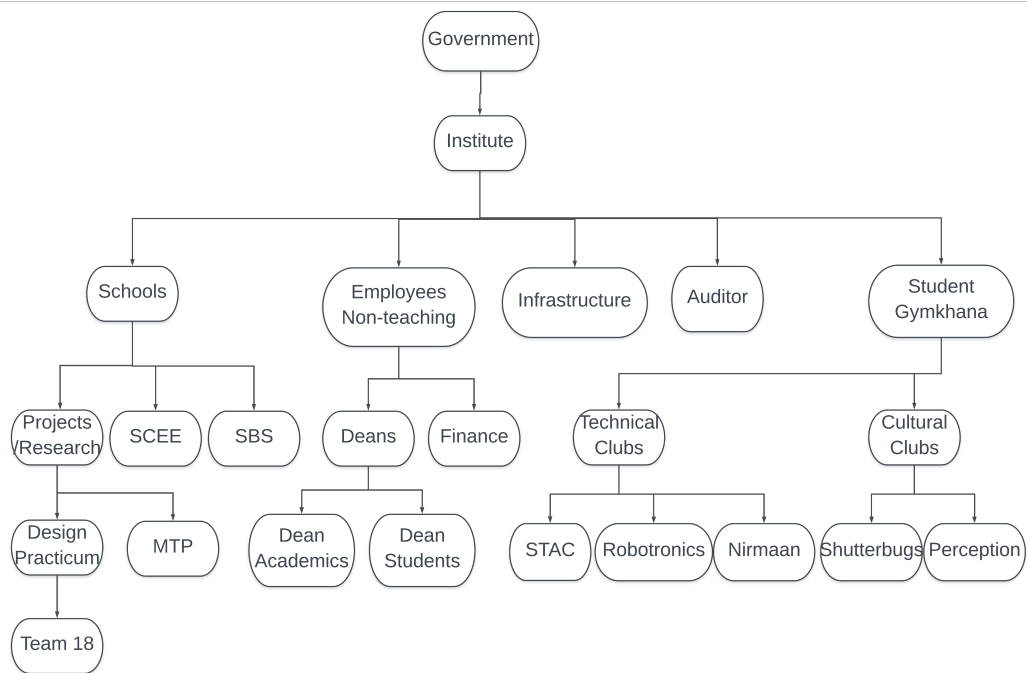
2.3   Information Modelling

Figure 2.1: The Hierarchy

Chapter 3

Conceptual Design

3.1 Entities

This section will explain the entities and their use. On our platform, every person is modelled as a user, and a user can have many roles. Hence, each has a personal account, which is maintained separate from the other accounts that he/she may manage.

3.1.1 Accounts

This entity is meant as a model of the different levels of the Institute Bodies. From the figure, we can observe a hierarchy based on the money flow. At the highest level, money is assumed to flow into the system from the Government. The outflow is basically the expenditure by the institute itself, and at the lower levels(constituted primarily by students). Lower level accounts can ask request for money from higher accounts. In the real world, this is just like asking money from a higher authority.

3.1.2 Users

This entity is to store the information of the the users of the institute. They could range from students to professors, and other employees. Important details such as bank details are captured here.

The reason accounts and users are separate is because the account can represent any body(such as SCEE and ShutterBugs), and they can be run by any user.

3.1.3 Transactions (Weak)

This entity is used to log the money flow in the system. This also helps in statistics, in transparency and to increase the speed of the system in general. These are initiated by higher level accounts to transfer some money to a lower level account.

3.1.4 Requests

These are exactly what they mean. A lower level account can request money/budget from higher accounts through this medium.

3.1.5 Bills (Weak)

To manage auditing of the lower level expenditures, this feature enables capture of such information to enable speedy and transparent auditing process. This is a weak entity, as it cannot exist without an Account. In other words, it is always dependent on an account for its existence.

3.2 Relations

3.2.1 useraccounts — User - Account

This relations captures the information regarding which user has access to which account. There is an additional attribute called Tag, that captures the information of the type of relation it is. For example, a user can be a Team Leader for a team project, and another user can be the coordinator for the parent account. This way, it is abstract so as to capture the maximum detail.

3.2.2 Parent-Child — Account-Account

The accounts that provide the money for a particular account are termed as its parents. The accounts that receive money are termed as children.

3.2.3 Expenditure — Bills-Account

This relation captures the information about the expenses of particular account.

3.3 The Special Account — Auditor

This account is specifically meant for auditing requests after verifying bills. This is called as a special account as it is only meant to handle approval or disapproval of requests. It was modelled as an Account, so that multiple people may log in as an Auditor. Whenever a request for money is made on the platform, the request is sent to the account. The admin who manages the account
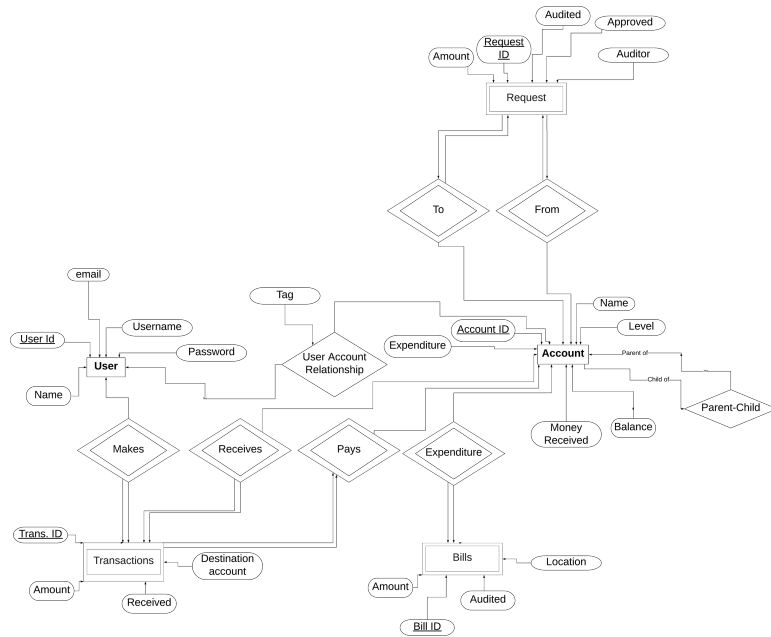
3.4 The E-R Model

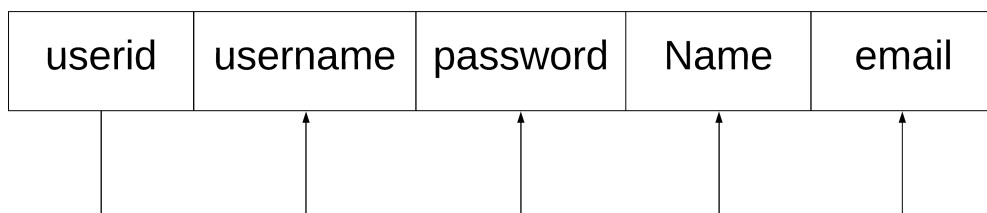Figure 3.1: ER Diagram
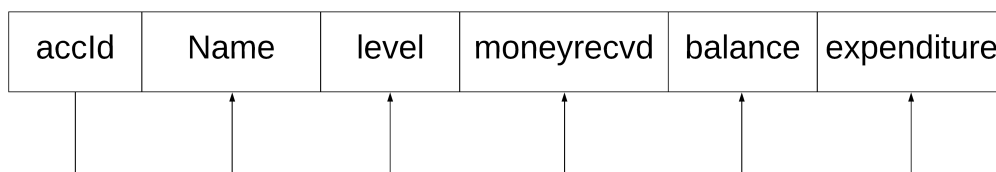
Chapter 4

Logical Schema

4.1   User

userid	username	password	Name	email

Figure 4.1: Entity User

4.2   Account

accId	Name	level	moneyrecvd	balance	expenditure

Figure 4.2: Entity Account

4.3   Relation User-Account

userId	accID	Tag

Figure 4.3: User-Account