项目名称：ShadowsTalks (ST)

目标：为用户提供一站式透明的访问各大SNS。

1. 用户在ST上注册并与多个不同的SNS帐号绑定。用户通过ST提供的统一UI从绑定的账户发送消息，接收消息。从ST用户角度看，他只通过ST的帐号在和别人沟通，但实际的消息是通过现有的SNS进行传播。
2. 用户通过ST可以组建私密组。消息的所有者(SNS上的楼主，po主等)可以设定密钥，然后通过ST提供的加密功能，将消息加密后发布到SNS上，并且可以创建和管理该消息的私密组，加入到该组的用户(ST用户)被授权通过密钥服务访问消息所有者的密钥，由ST解码后将消息展现给该组用户，并且组内成员转发或评论的内容必须使用该密钥加密。

项目步骤

1. 实现ST用户通过ST绑定多个SNS帐号(weibo,renren, etc…)进行消息发布。
2. SNS帐号的管理。
3. 将各种SNS平台的抽象成统一服务，为用户提供统一的接口。
4. ST 的UI设计。
5. 消息的发布、接收、查看和删除等。

二. 实现ST对ST用户的管理

1. 设计用户之间的关系(每个ST绑定不同的帐号，ST用户关注另一个ST用户需要映射到其对应的SNS帐号)。

2. 查找用户。

3. 添加关注。

4.

三. 实现私密消息

1. 对要消息加密和解密。

2. 对消息创建私密组。

3. 构建密钥访问系统。

4. 私密组成员管理。

5. 其他用户通过私密组转发评论消息。

6.

项目模块

一. SNS的抽象模块

(微博：转发，评论等功能；微信：xxx，xxx功能)

将这些功能统一提供给ST

将消息抽象成*Talks*,将加密消息抽象成*shadowTalks*

二. 加密模块

输入：消息字符，密钥，etc.

输出：加密字符

格式： ShadowTalks\_*xxx*\_[!!!]\_???\_*hashCode*

其中*xxx*为ST用户名，！！！为选项，???为加密消息，*hashCode*为前面字符串的哈希值，用于访问密钥服务。每个*hashCode*对应一个加密消息和私密组。

三. 密钥服务模块

管理发布者，*hashCode*，密钥，私密组成员的信息。

四. ST平台UI

**模块连接**

M3

Talks

Strings

M1

M4

User

M2

server

follower

key\_resp

Y

Decode

show

is *ST*

browse *SNS*

key\_req

key\_req

key\_req

add\_access

add\_foo

foo\_req

access\_req

connect to public SNS

connect to public SNS

post *Talk* with *key.*

put hashCode, key, etc.to server

publisher

set access member

密钥服务