4347 Database Systems: Project Proposal

**Team Members / NetID:**

* Olumayowa Labinjo
  + ool180000
* Po-Yu Liu
  + pxl170130
* Jack Zumwalt
  + jsz160130
* Johnny Jiang
  + hxj170003

**Project Proposal:**

* Our team will create a platform to back-test cryptocurrency trading strategies using data from a cryptocurrency fear & greed index API.
* The program will calculate different market indicators and show when the best time to buy and sell is.
* Our group will be able to CRUD the contents of the data, managing it as necessary to provide users with accurate data and filters
* API: <https://alternative.me/crypto/fear-and-greed-index/>
* This platform will not only be able to backtest cryptocurrencies but also a variety of different types of derivatives contracts.
* This will include stock options and futures contracts. We should be able to find the data needed for the options strategies by using the TDameritrade or Yahoo API.
* As for the futures contracts, all the information we need should be provided by the CME group. They also have several APIs depending on the exact projects your working on.
* All the information will be funneled through to the program creating several different types of market indicators depending on what you trade and your trading goals.
* CME Group API: <https://www.cmegroup.com/market-data/datamine-api.html>
* TDameritrade API: <https://developer.tdameritrade.com/apis>

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='TRADITIONAL,ALLOW\_INVALID\_DATES';

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci ;

USE `mydb` ;

-- -----------------------------------------------------

-- Table `mydb`.`Broker`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `mydb`.`Broker` (

`broker\_id` INT NOT NULL,

`order` TINYINT NULL,

`amount` FLOAT NULL,

`asset\_name` VARCHAR(45) NULL,

PRIMARY KEY (`broker\_id`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `mydb`.`Customer`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `mydb`.`Customer` (

`customer\_id` INT NOT NULL,

`order` TINYINT NULL,

`amount` FLOAT NULL,

`ssn` INT NULL,

`Broker\_broker\_id` INT NOT NULL,

PRIMARY KEY (`customer\_id`),

INDEX `fk\_Customer\_Broker1\_idx` (`Broker\_broker\_id` ASC),

CONSTRAINT `fk\_Customer\_Broker1`

FOREIGN KEY (`Broker\_broker\_id`)

REFERENCES `mydb`.`Broker` (`broker\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `mydb`.`Trade History`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `mydb`.`Trade History` (

`trade\_id` INT NOT NULL,

`order` TINYINT NULL,

`amount` FLOAT NULL,

`asset\_name` VARCHAR(45) NULL,

`time` DATETIME NULL,

`Customer\_customer\_id` INT NOT NULL,

PRIMARY KEY (`trade\_id`),

INDEX `fk\_Trade History\_Customer1\_idx` (`Customer\_customer\_id` ASC),

CONSTRAINT `fk\_Trade History\_Customer1`

FOREIGN KEY (`Customer\_customer\_id`)

REFERENCES `mydb`.`Customer` (`customer\_id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `mydb`.`Asset`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `mydb`.`Asset` (

`asset\_id` INT NOT NULL,

`price` FLOAT NULL,

`volume` FLOAT NULL,

`asset\_name` VARCHAR(45) NULL,

PRIMARY KEY (`asset\_id`),

CONSTRAINT `asset\_name`

FOREIGN KEY ()

REFERENCES `mydb`.`Broker` ()

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;