系統設計書

System Design Document

目錄

1. [修訂版本記錄](#_TOC_250040) 5
2. [簡介](#_TOC_250039) 6

[目標](#_TOC_250038) 6

[產品概觀](#_TOC_250037) 6

[範圍](#_TOC_250036) 6

1. [開發環境](#_TOC_250034) 7

[Development tools](#_TOC_250033) 7

[Development standards](#_TOC_250032) 7

[System environment](#_TOC_250031) 7

1. [系統架構](#_TOC_250030) 8

[Architecture diagram.](#_TOC_250029) 8

[Determines Integration Procedure/Sequence](#_TOC_250028) 9

[System Design Constraints](#_TOC_250027) 9

1. [子系統設計](#_TOC_250026) 10

[子系統 1](#_TOC_250025) 10

[使用者介面描述](#_TOC_250024) 10

畫面 10

物件及動作 10

元件 1 設計規格 11

元件 1 介面描述 11

元件 1 運作說明 11

互動圖 11

元件 2 設計規格 12

元件 2 介面描述 12

元件 2 運作說明 12

互動圖 12

元件 3 設計規格 13

元件 3 介面描述 13

元件 3 運作說明 13

互動圖 13

元件 4 設計規格 14

元件 4 介面描述 14

元件 4 運作說明 14

互動圖 14

[元件整合條件](#_TOC_250023) 14

[子系統 2](#_TOC_250025) 15

[使用者介面描述](#_TOC_250024) 15

畫面 15

物件及動作 15

元件 1 設計規格 16

元件 1 介面描述 16

元件 1 運作說明 16

互動圖 16

[元件整合條件](#_TOC_250023) 16

[子系統 3](#_TOC_250025) 17

[使用者介面描述](#_TOC_250024) 17

畫面 17

物件及動作 17

元件 1 設計規格 18

元件 1 介面描述 18

元件 1 運作說明 18

互動圖 18

元件 2 設計規格 19

元件 2 介面描述 19

元件 2 運作說明 19

互動圖 19

[元件整合條件](#_TOC_250023) 19

1. [外部介面設計](#_TOC_250020) 20

[軟體介面設計](#_TOC_250019) 20

[設計規則](#_TOC_250018) 20

1. [非功能設計](#_TOC_250010) 21

[效能設計](#_TOC_250009) 21

[安全性設計](#_TOC_250008) 21

[可得性設計(Availability Design)](#_TOC_250007) 21

[可靠性設計(Reliability Design)](#_TOC_250006) 21

[可維護性設計(Maintainability Design)](#_TOC_250005) 21

1. [資料設計](#_TOC_250004) 22

[內部軟體資料結構](#_TOC_250003) 22

[暫存性資料結構](#_TOC_250001) 22

[資料庫描述](#_TOC_250000) 22

# 修訂版本記錄

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修改日期 | 說明 | 修改者 |
| 1.0 | 2018/5/19 | 初步版本 | 司徒華傑 |
| 1.1 | 2018/5/24 | 校正 | 司徒華傑 |

# 簡介

## 目標

## 專案目標

## 鑒於現在社會大眾對農作產業越來越不了解，本團隊希望能夠以系統模擬的方式，寓教於樂，讓使用者能夠透過本團隊開發的系統對農作產業有更進一步的認識，了解農作產業經營的辛苦。

## 設計目標

## 期望以物件導向進行專案程式設計，降低程式碼之複雜度，清楚的專案程式架構，使專案成員方便程式撰寫及日後專案之新增、修改及維護。

## 產品概觀

## 本專案產品為文字化界面運作之模擬農作物栽培監控系統遊戲，功能包含植株之選擇栽培、澆水、施肥、病蟲害防治、成本經營等，並包含模擬台灣農業之季節氣候與植株特性等要素，使產品模擬之環境貼近真實農業之狀況。

## 範圍

## 產品試用使用者範圍廣泛，老少咸宜。

## 透過此產品之使用，能使成年使用者更加了解農產業之辛苦，進而珍惜資源不浪費，也期盼年幼之使用者透過此產品對農業產生興趣，未來投身台灣農產業，成為台灣之農業人才。

# 開發環境

Development tools  
notepad++、eclipse

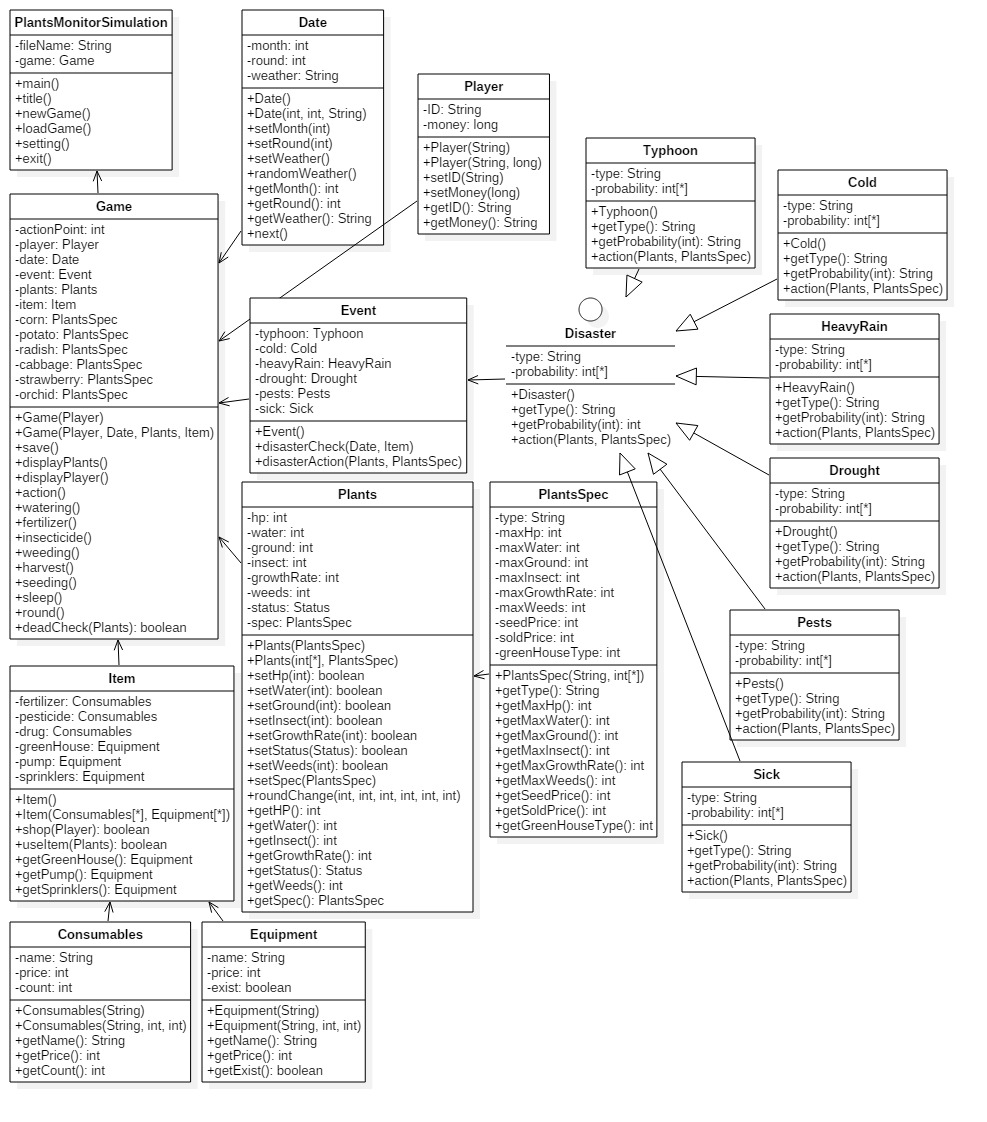
Development standards  
JAVA 1.8.0 JDK、JAVA 1.8.0 JRE

System environment

Windows 10

# 系統架構

Architecture diagram



Determines Integration Procedure/Sequence

1. Date、Player、Consumables、Equipment、PlantsSpec
2. Plants
3. Item、Disaster
4. Event
5. Game
6. PlantsMonitorSimulation

System Design Constraints

複數詞組成之名稱，其每個詞開頭皆為大寫，中間不以底線做連結

Class名稱首字大寫

變數名稱首字小寫，所有變數皆訂為private

做為系統設定的非變數設為常數，名稱為全大寫並以底線連接複數詞

Function首字小寫

每個Class須有一至兩個constructor，其一只需輸入必要參數以供建立全新物件之使用，另一個須輸入所有參數作為讀取遊戲時回復物件之使用

# 子系統設計

## 子系統 1

Game

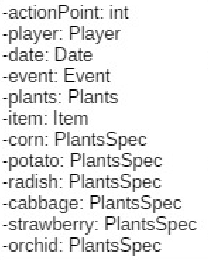
### .使用者介面描述

進入遊戲後負責遊戲回合之運行

. 畫面



. 物件及動作

物件 動作

元件 1 Date 設計規格

負責遊戲系統時間之設定控制

. 元件 1 介面描述

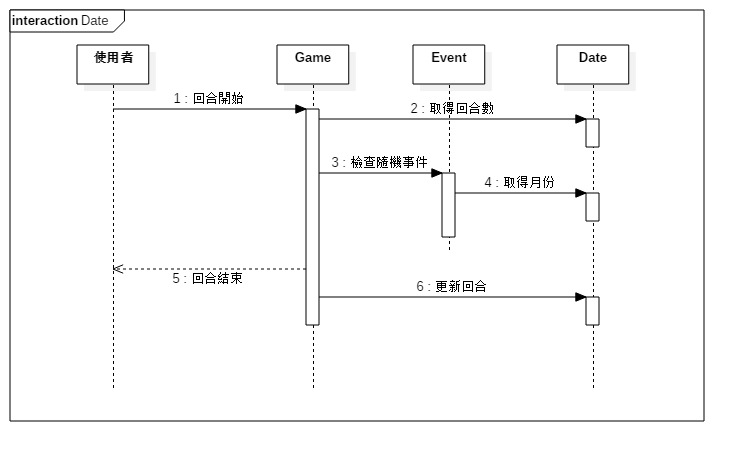
以文字呈現於遊戲畫面之上方偏右位置

表現月份及回合數

. 元件 1 運作說明

回合初開始輸出目前月份及回合數，動作結束後回合數加一，四回合後回合數重設為一，月份加一，月份至十二月後月份重設為一月。

. 互動圖



元件 2 Player 設計規格

負責使用者之ID及金錢

. 元件 2 介面描述

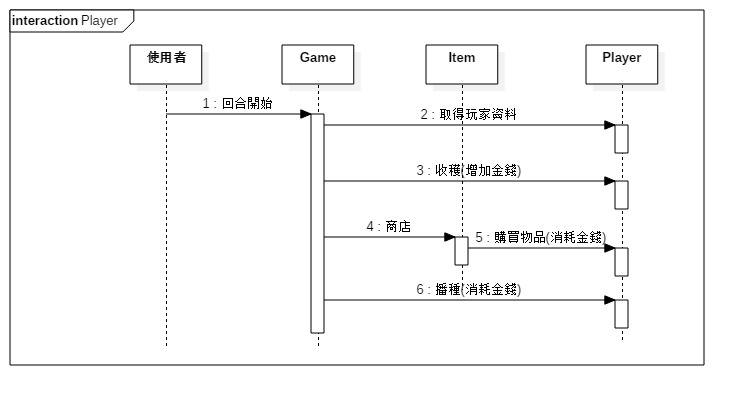
以文字呈現於畫面右上角位置

表現使用者之ID及金錢數目

. 元件 2 運作說明

接收到主程式傳送之使用者ID以及使用者擁有之金錢數目後，輸出信息於畫面上

. 互動圖



### 元件 3 Plants 設計規格

### 負責管理植物的各項參數

### .元件 3 介面描述

### 於螢幕上顯示目前栽種植物之HP、水分、肥沃度、蟲害、雜草量、成長進度及狀態

### .元件 3 運作說明

### 若為讀取舊檔，則接收主程式傳送之植物參數紀錄檔後輸出於螢幕上

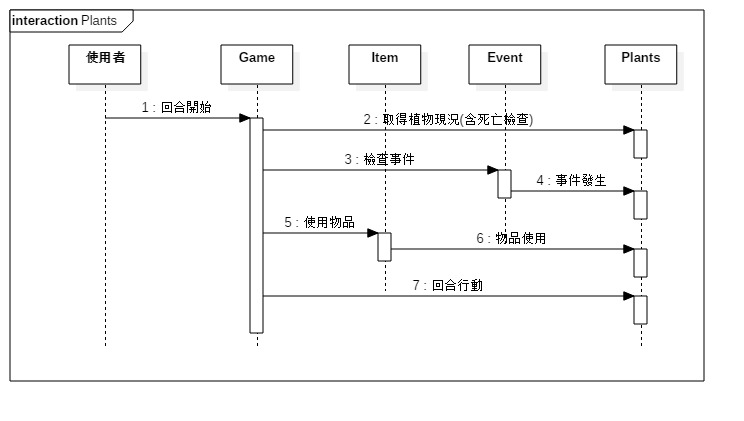
### 若為開新檔案，則建立植物之原始參數並輸出於螢幕上

### 回合開始時根據天氣及事件更新各項參數

### 使用者操作動作後根據使用者之操作更新各項參數

### 

### .互動圖



### 元件 4 PlantsSpec 設計規格

### 負責管理植物種類及參數最大值

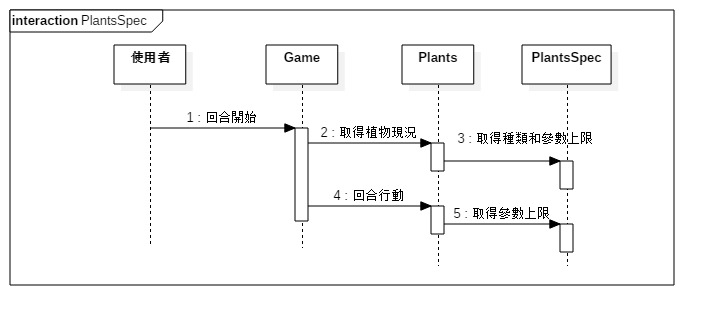
### .元件4 介面描述

### 根據不同種類之植物，設定輸出於螢幕上之各種參數最大值

### .元件 4 運作說明

### 根據使用者選擇之植物將相對應之參數最大值傳給Plants運用

### .互動圖



### 元件整合條件

### Game中會有一個Date、一個Player、數個於程式執行時已被確

### 定且不會變動的PlantsSpec，以及一個會隨遊戲運行不斷被替換

### 的Plants

### 子系統 2

### Event

### .使用者介面描述

### 負責每回合開始之隨機事件安排

### .畫面

### 

### .物件及動作

### 

### 物件 動作

### 元件 1 Disaser 設計規格

### 負責各個災害的運行控制

### .元件 1 介面描述

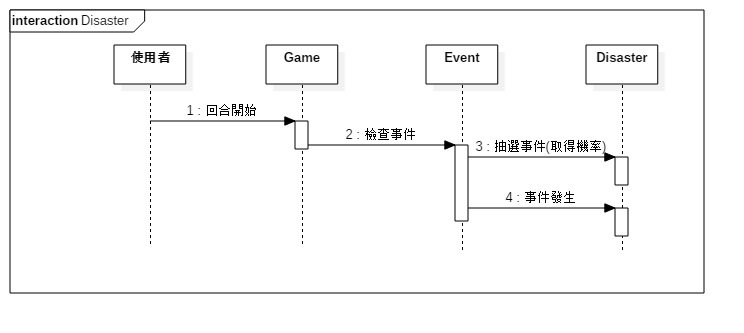
### 將每月隨機產生之事件災害顯示於螢幕上

### .元件 1 運作說明

### 每月開始隨機產生0~1個事件災害

### 配合事件災害改變植物之參數及狀態

### .互動圖



### 元件整合條件

### Event中會有一個Disaster作為Interface並有遊戲執行前便已確定的

### 數樣災害種類

### 子系統 3

### Item

### .使用者介面描述

### 負責商店、消耗品以及設備之管理

### .畫面

### 

### .物件及動作

### 

### 物件 動作

### 元件 1 Consumables 設計規格

### 負責消耗品之管理

### .元件 1 介面描述

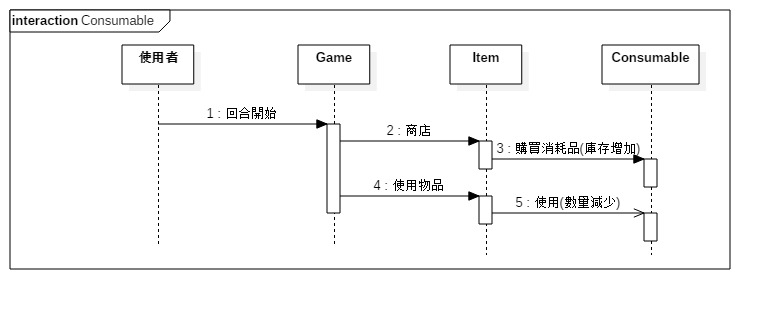
### 於商店及倉庫顯示消耗品之價格、名稱及數量

### .元件 1 運作說明

### 將使用者於商店內購買之消耗品新增至倉庫以供使用

### 使用時根據不同消耗品特性修改植物之各種參數

### .互動圖



### 元件 2 Equipment 設計規格

### 負責設備之管理

### .元件 2 介面描述

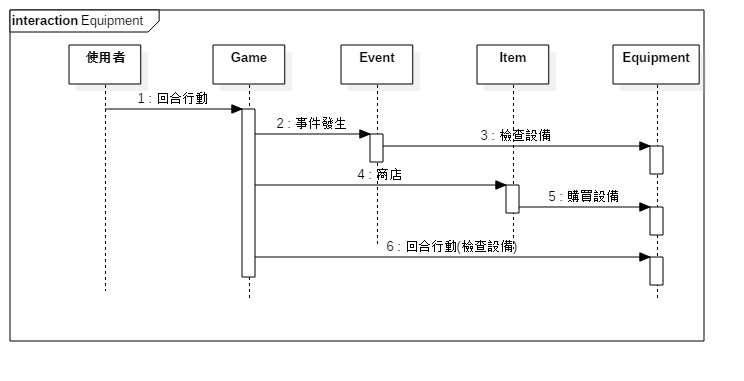
### 於商店及設備選項顯示設備之價格及名稱

### .元件 2 運作說明

### 將使用者於商店購買之設備新增於設備選項並啟用其功能

### 根據不同設備之特性調整植物之各項參數

### .互動圖



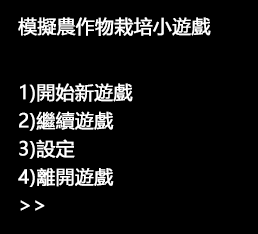
### 元件整合條件

### Item中會有數樣已被確定的Consumables和Equipment，並且只能

### 改變該消耗品的數量和該設備的啟用與否

# 外部介面設計

## 軟體介面設計



### 設計規則

### 主介面要有新遊戲及讀取舊遊戲檔案兩項目

### 新遊戲必須詢問使用者ID，將ID和初始金額建成Player後傳進Game中

### 讀取舊遊戲則需要讀取記錄中的使用者ID及金錢餘額、月份及

### 回合、植物現況、道具及設備購買狀況並分別建成物件後傳入Game中

### 

# 非功能設計

## 效能設計

## 將每一種植物種類分類於PlantsSpec並用其記錄每一種參數之上

## 限，如此在種植新植物時只需要根據該植物的種類導入對應的

## PlantsSpec，不需要再一一傳入所有該植物的參數上限。

## 安全性設計

## 將所有變數皆訂為private，對使用者ID進行檢查(限定字數及限定英文及數字)

可得性設計(Availability Design)

讓使用者只需輸入代號做為行動之選擇以減少使用者輸入的複

雜度及錯誤

可靠性設計(Reliability Design)

結構簡化

每回合皆做自動存檔，以防意外造成系統關閉時所造成的影響

可維護性設計(Maintainability Design)

將每項災害分別建成不同的類別，藉此便於維護及調整不同的

災害，另外也將作為遊戲設定的非變數設為常數以便調整

# 資料設計

## 內部軟體資料結構

## Event、Disaster、PlantsSpec

## 暫存性資料結構

## Action Point

## 資料庫描述

## Player、Date、Plants、Item

## 以txt檔存放以上四種物件的參數