

2021-協同產品設計實習-stage2-ag12 籃球機

組長:40823152

組員:40823153、40823110、40823125

產品目標

主題:籃球機 二代

動機:

第一次小組產品的籃球機只能自己轉動無法控制，所以二代的籃球機目標增加案件控制轉動。

製作規劃:

w5:確認主題，工作分配

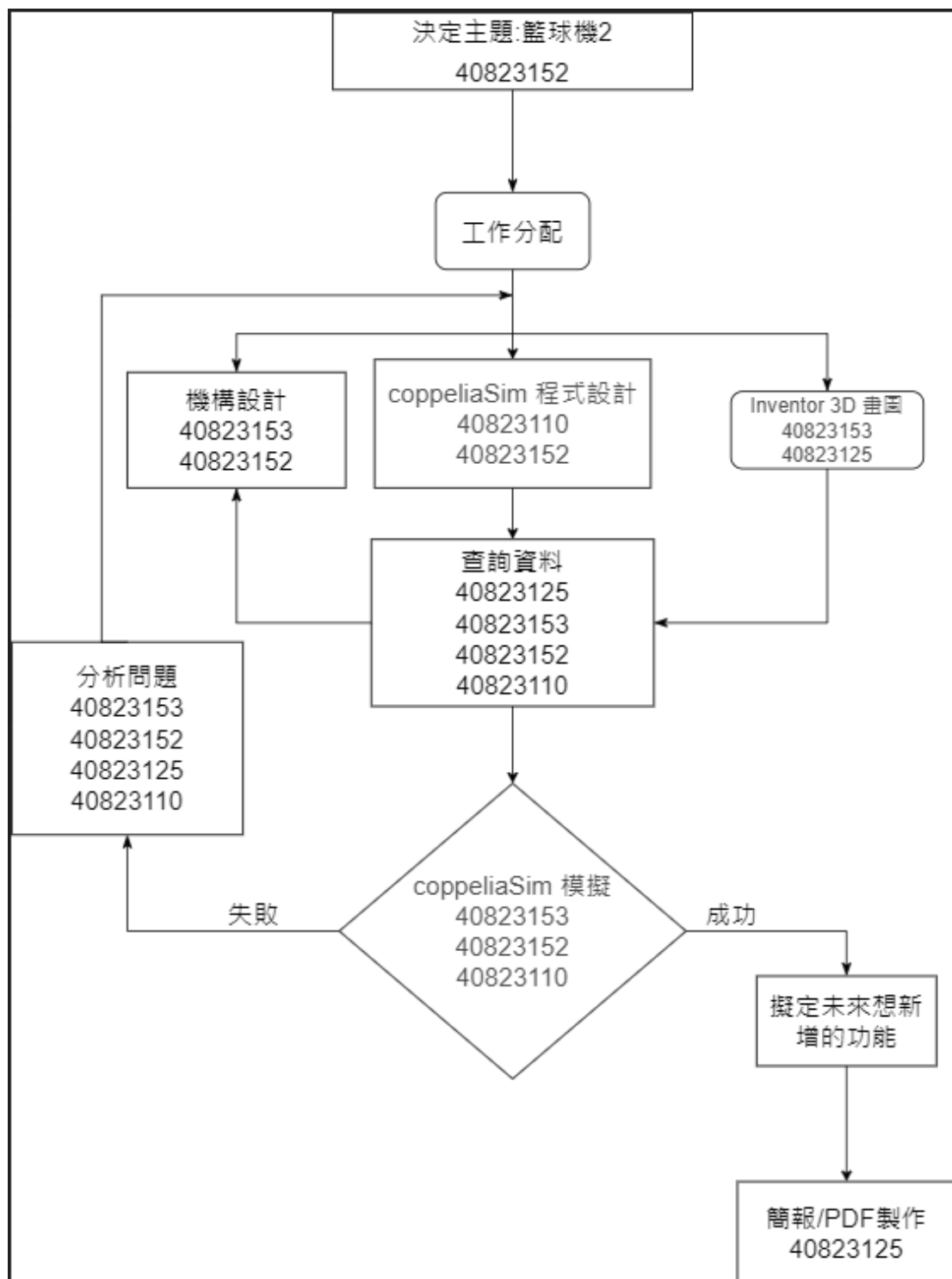
w6:尋找 lua 程式與 coppeliaSim 文章，並且使用 lua 程式控制軸在 coppeliaSim 裡模擬，籃球機外觀修整。

w7:完成按鍵控制程式與模擬，籃球機外觀修整。

w8:除錯與修整，完成簡報與 pdf

w9:報告

工作流程

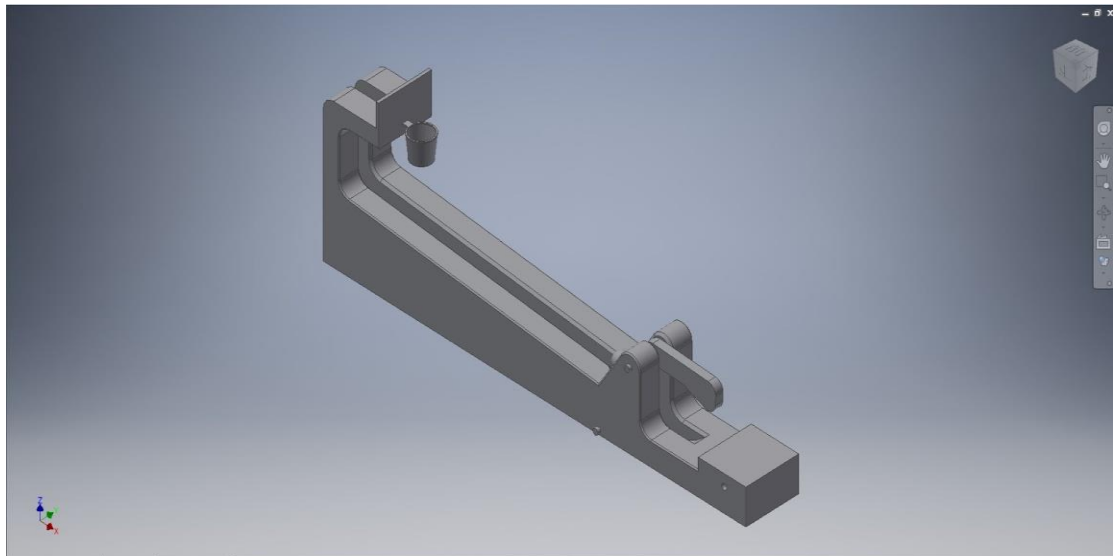


一代版本四

籃球機 4: <https://drive.google.com/drive/folders/1GhgbMJbaKh2e0Zb0xjIF-sdGHJVy5C8D?usp=sharing>

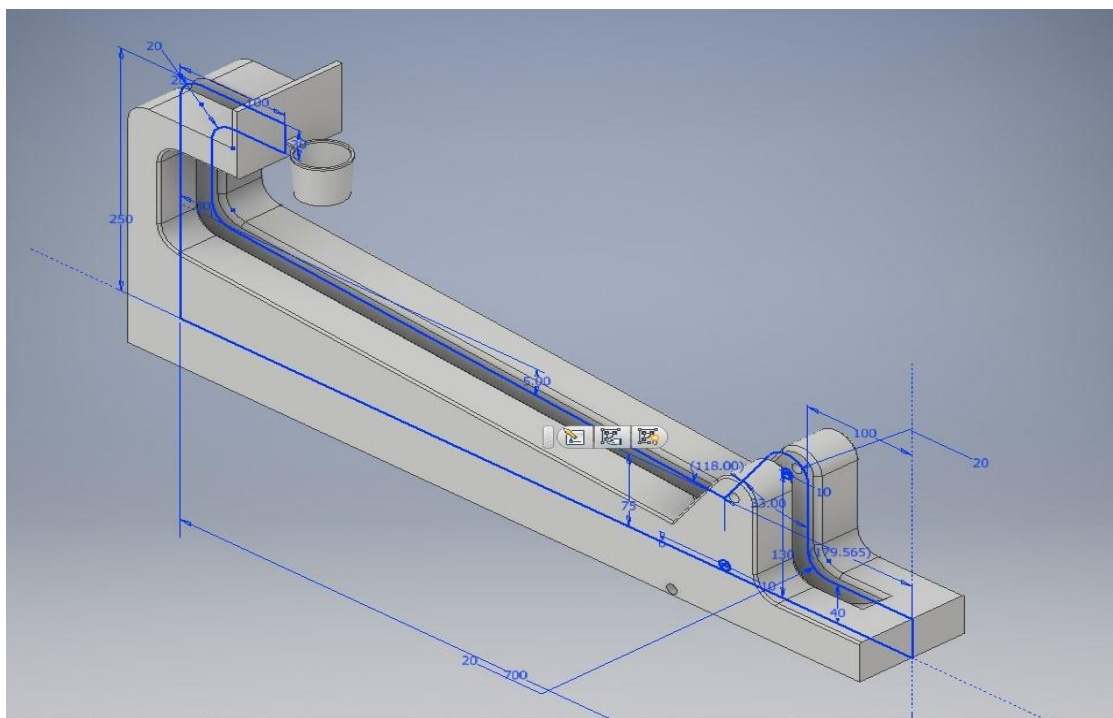
coppeliaSim 模擬-版本 4:<https://drive.google.com/file/d/1vWfztZrbkWklqTl-B5sKLRIM5YI0OJST/view?usp=sharing>

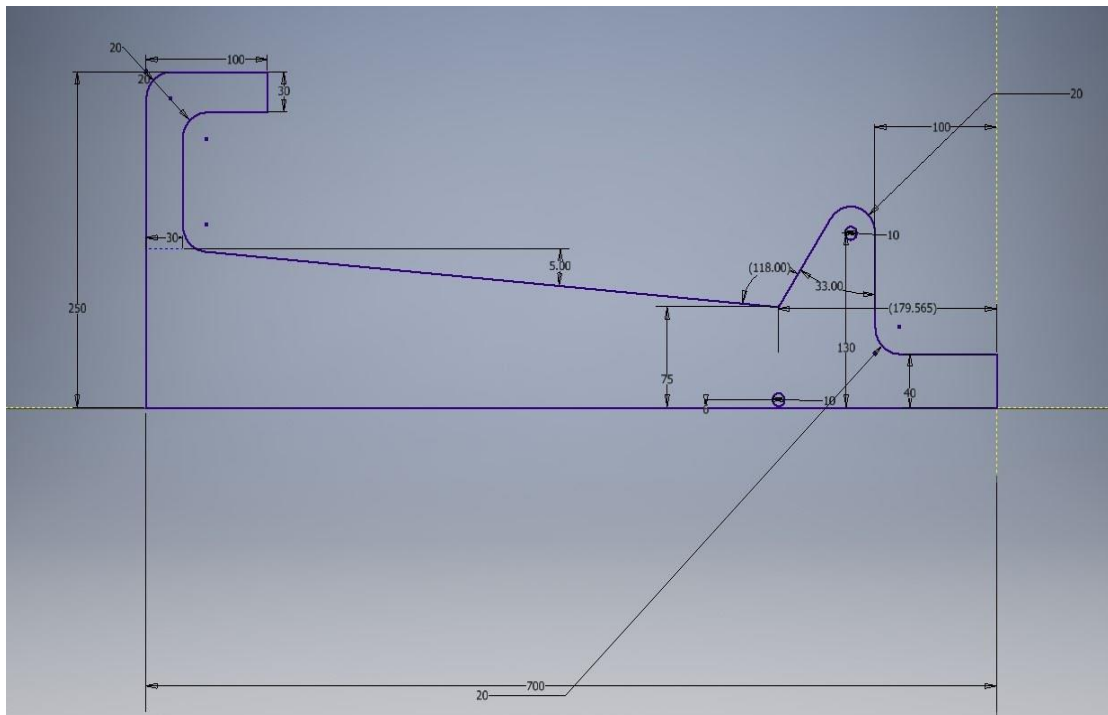
因為版本 2 在 coppeliaSim 裡被自動放大 10 倍，所以版本 3 無法擊中籃框，所以在 coppeliaSim 裡版本 4 放大 10 後就能擊中籃框。



因為版本 2 在 coppeliaSim 裡被自動放大 10 倍，所以版本 3 無法擊中籃框，所以版本 4 放大 10 後就能擊中籃框。

可調性





影片:<https://www.youtube.com/embed/y-ztP91swPs>

檢討

- 1.球進入籃框後回到擊球處的路徑不穩。
- 2.擊球桿只能轉動，無法控制出擊時間。
- 3.簡報內容不詳細
- 4.沒有理論依據

解決方法:

- 1.要新增球軌道的循環系統
- 2.使用 lua 控制軸
- 3.要有針對內容去延伸介紹為什麼要這麼做
- 4.使用數學得正答案。 如:使用能量守恆去計算速度和高度。

Lua control speed

使用 Lua 程式控制 coppeliaSim 擊球桿的轉速

影片:<https://www.youtube.com/embed/4xGjKBDXPFs>

keyboard control

使用 Lua 程式寫出可以由鍵盤按鈕控制 coppeliaSim 的擊球桿

影片:<https://www.youtube.com/embed/A5Ws3gGDpic>

Inquire keyboard number

查詢 keyboard 按鍵

```
print(auxiliaryData[1])  
    if (auxiliaryData[1]==string.byte(' ')) then
```

介紹:

可以查詢鍵盤代碼

影片:<https://www.youtube.com/embed/4xye0RDVC8Y>

Lua keyboard Turn back

擊球桿新增後轉功能，可以有效控制擊球的時機。

影片:<https://www.youtube.com/embed/R4FlyPIJxqw>

參考資料

Remote API functions (Lua)

<https://www.coppeliarobotics.com/helpFiles/en/remoteApiFunctionsLua.htm>

Regular API reference

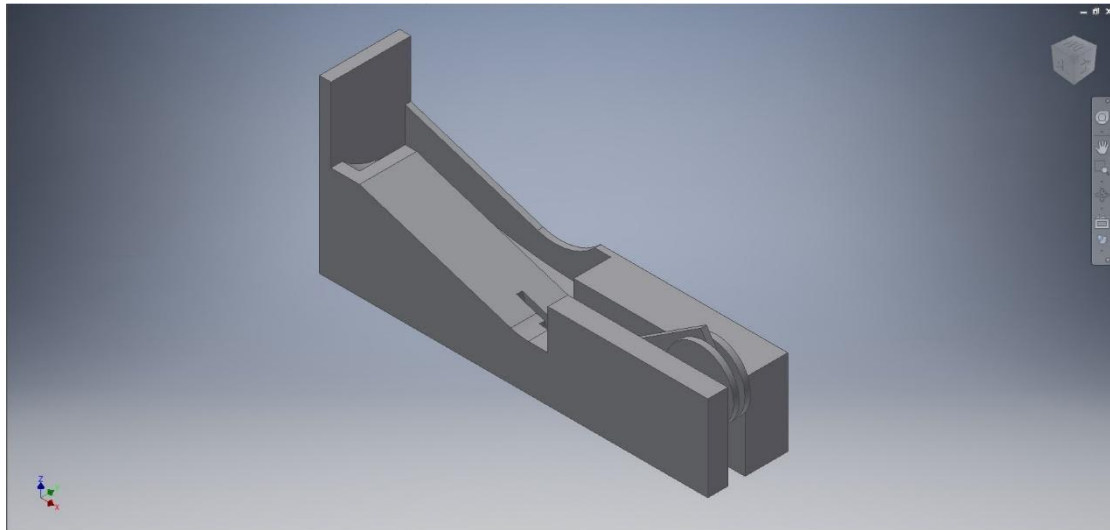
<https://www.coppeliarobotics.com/helpFiles/en/apiFunctions.htm>

W5

籃球機 2.1

<https://drive.google.com/drive/folders/1wxw3JJsXfnXkaYdWfED0NG7M9g2PI26I?usp=sharing>

設計緣由:想藉由曲柄連桿擊打球,方便 coppeliaSim 模擬與修正

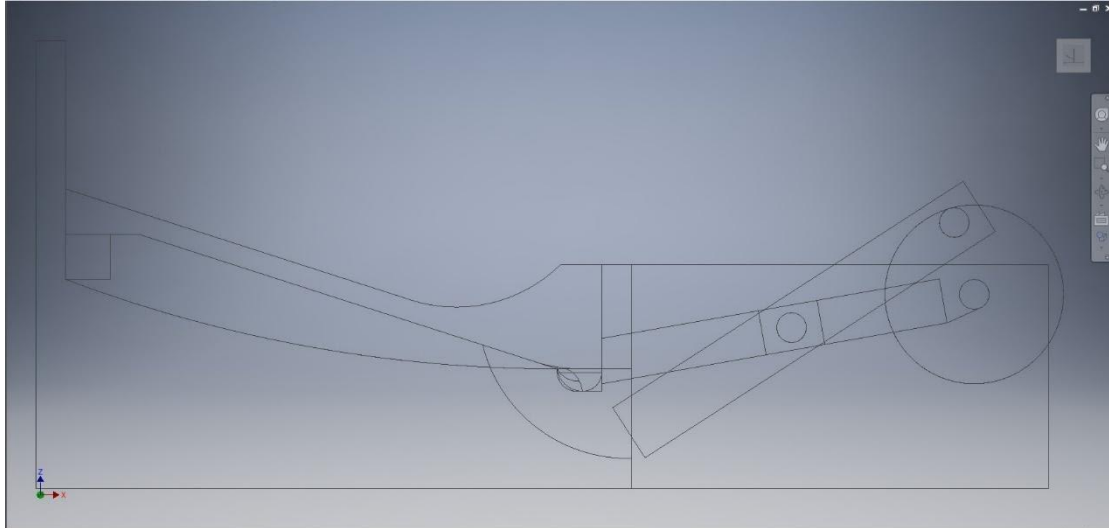


連桿機構的特點：

優點：運動副單位面積所受的壓力小且面接觸受力小，便於潤滑，磨損小；製造方便。

缺點：設計複雜誤差大。工作效率低。

原文網址：<https://kknews.cc/news/3l5m3ey.html>



主要問題:

- 1.過於簡陋
- 2.擊球桿會撞到基座

改善方法:

- 1.擊球桿的形狀
- 2.滑塊的距離縮短

W6

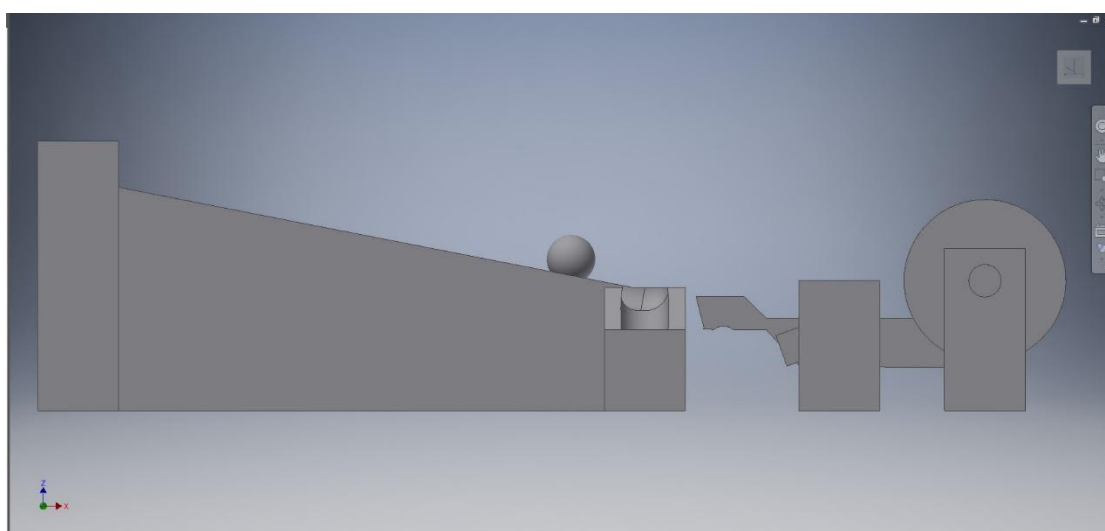
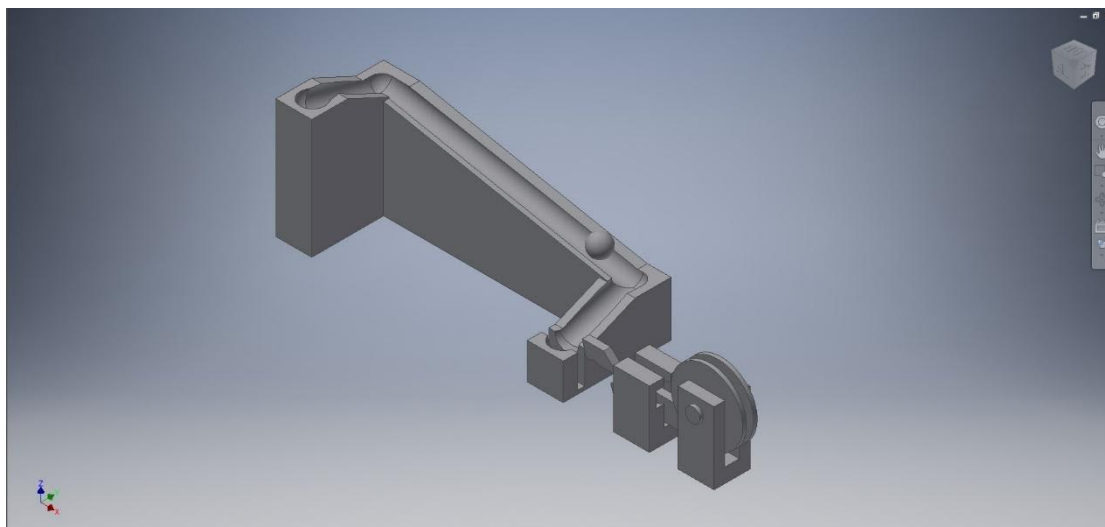
籃球機 2.2

<https://drive.google.com/drive/folders/1zqnQdyXmb-uHGltAk5ISACNNUaOuthIN?usp=sharing>

無框版

設計緣由:

- 1.想將擊球點角度提高,所以將擊球桿形狀設計的像車刀一樣。
- 2.為了讓擊球時滑塊的支點不會滑動,而且球掉落在頂端時有停頓的時間



影片

<https://youtu.be/fPKHOBXYZwY>

主要問題:

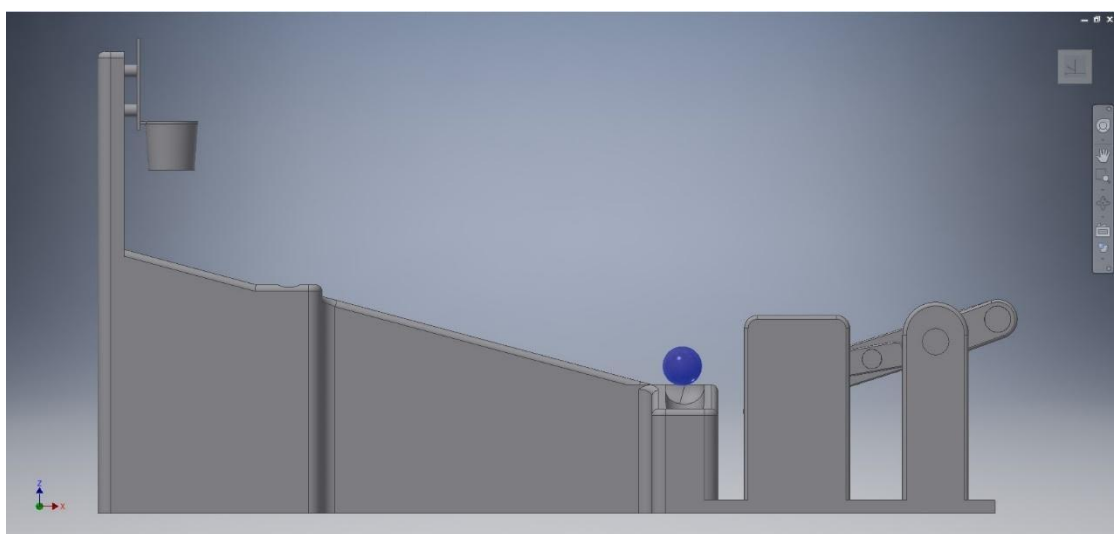
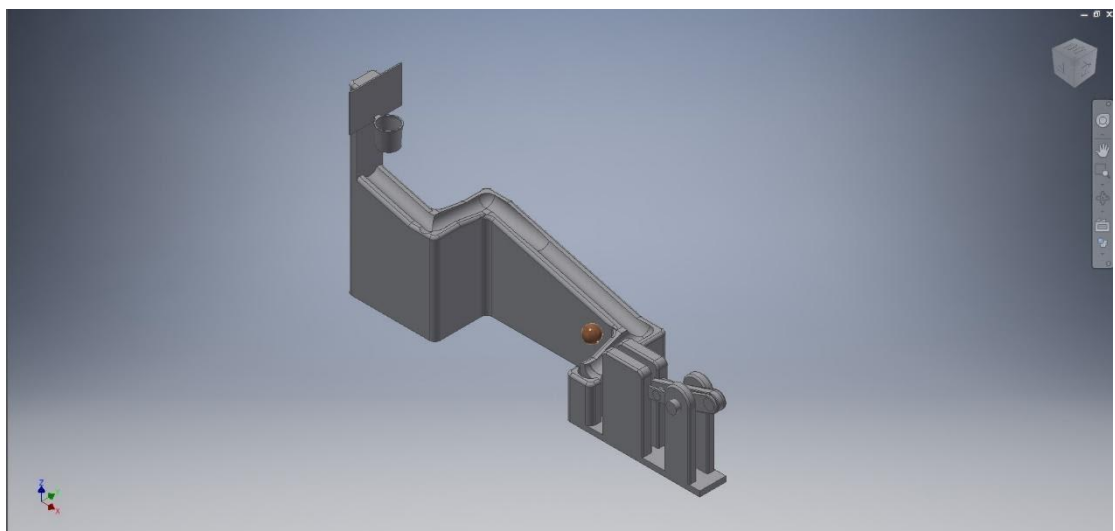
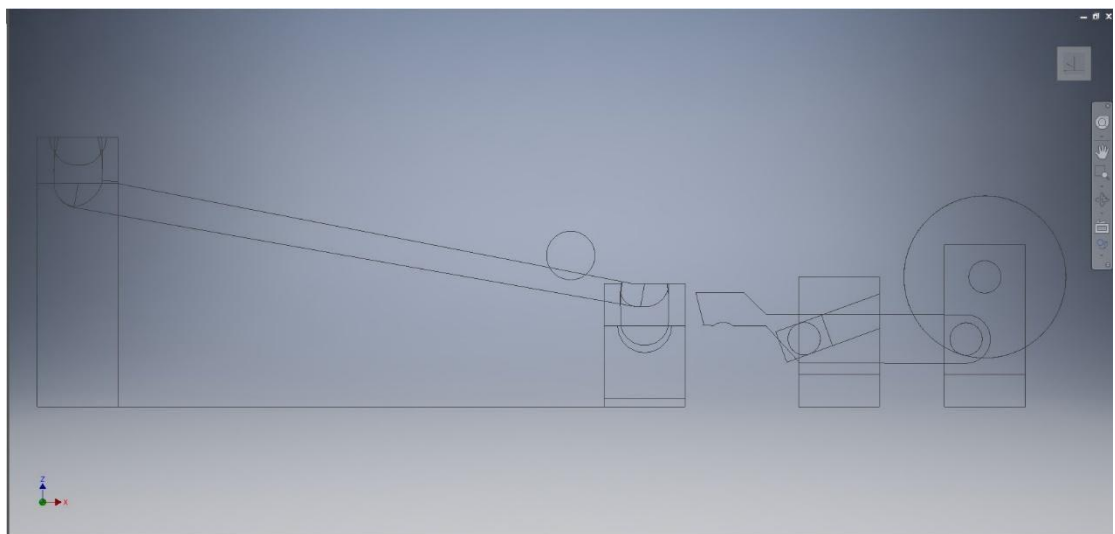
1.滑塊會撞到曲柄

改善方法:

1.擊球桿改為直的

有框版

設計緣由:為了讓滑塊不會撞到曲柄,所以將擊球桿改為直的



主要問題:

1.球模擬時不一定是直線

改善方法:

1.所以將斜坡面積變大

W7

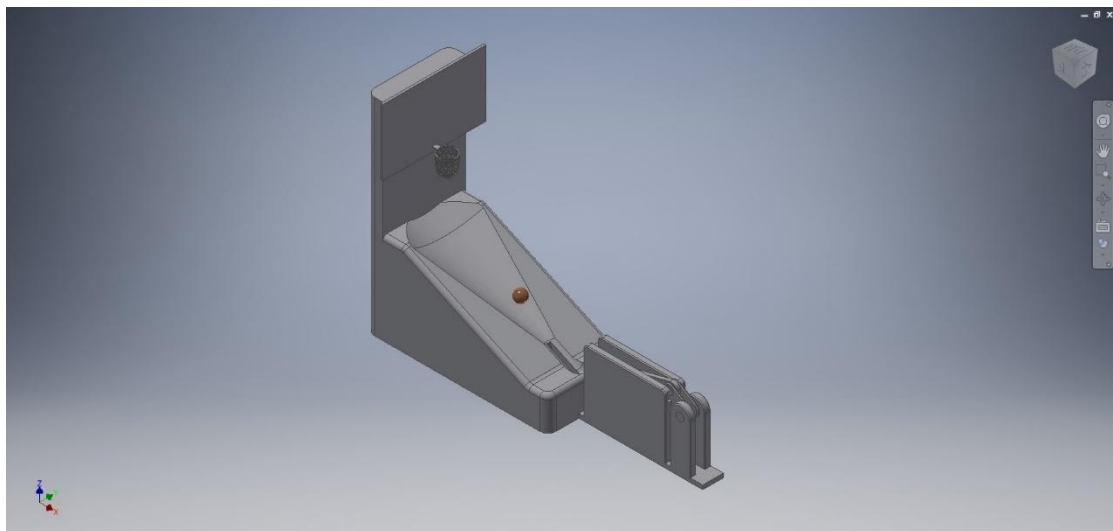
籃球機 2.3

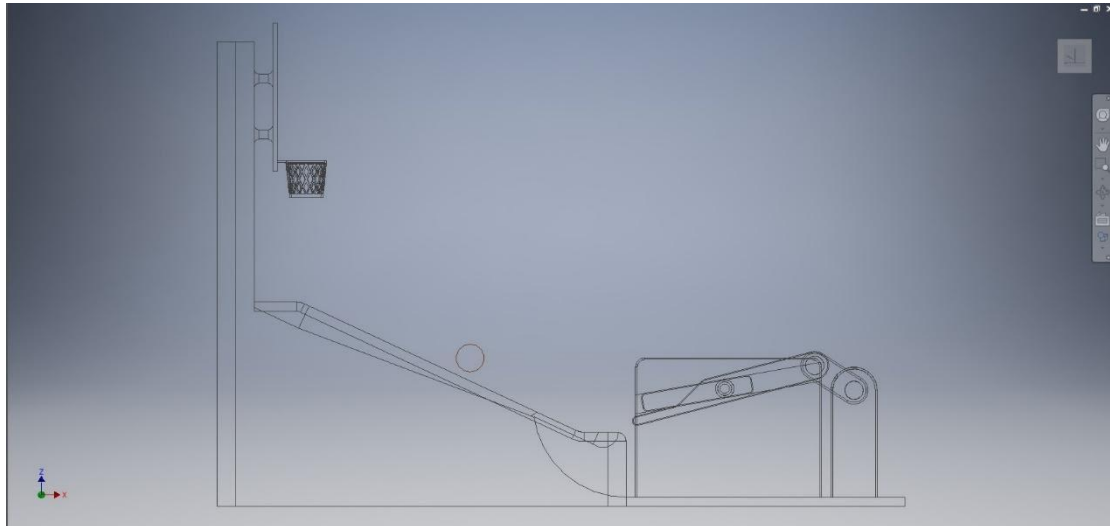
<https://drive.google.com/drive/folders/1gVL4FmbeqfpH1UVUGS85cwdQSOvvy683?usp=sharing>

設計緣由:

1.為了確定可以完整的接到球,所以將斜坡面積變大

2.為了讓滑塊不會撞到曲柄,所以改為曲柄滑塊機構





主要問題:

1.斜面的接球太斜而且不夠深,會導致球掉出來

2.CoppeliaSim v-rep 連桿運動問題

一開始在 CoppeliaSim 上模擬發現模型無法順利地運轉

<https://youtu.be/kpUwzR1XGkY>

改善方法:

1.將斜面挖深,並且將斜面改為單斜而不是曲型

2.CoppeliaSim v-rep 連桿運動問題 解決

後來才發現原來是尺寸的問題

由於 coppiasim 對與尺寸可能無法到 inventor 那麼精準所以要預留空隙,才能順利運轉

<https://youtu.be/97fUChcmisk>

W8

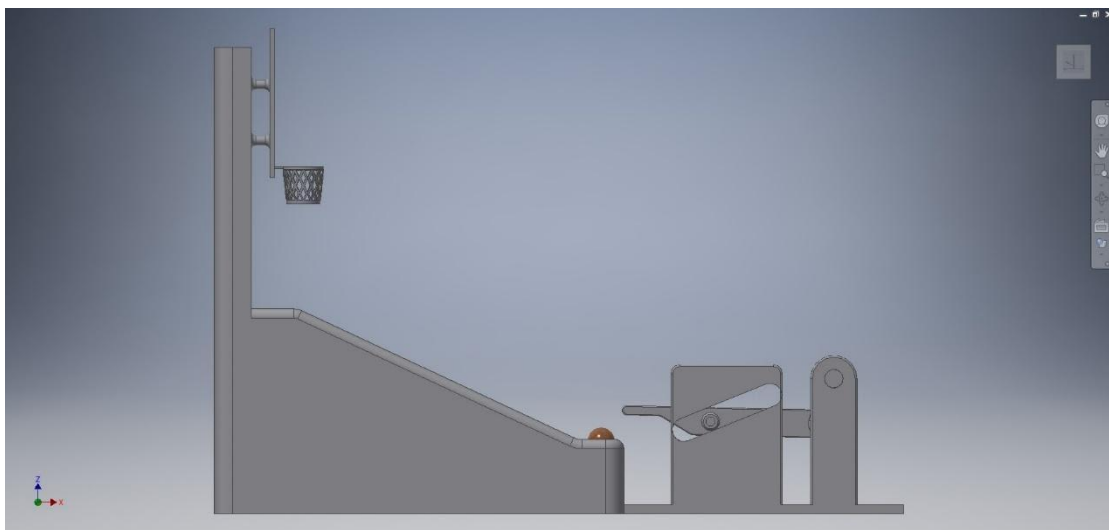
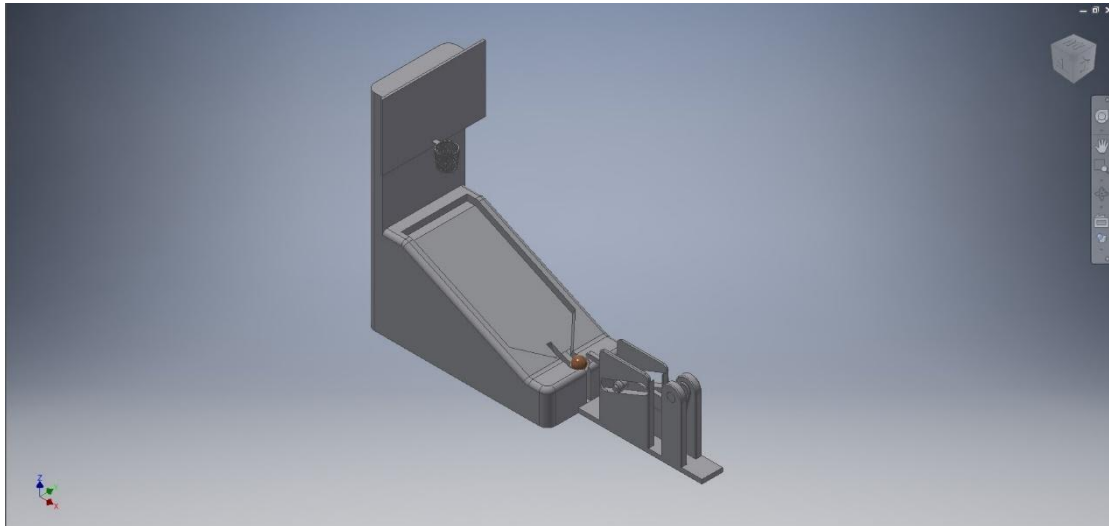
籃球機 2.4

<https://drive.google.com/drive/folders/1CCFrCBzbBhC9gnwaUUMYi5Cf7kd8Xkcz?usp=sharing>

2.4.1

設計緣由:

- 1.為了確定可以完整的接到球,所以將斜坡面積挖深並改為單斜面
- 2.為了滑動流暢將滑塊改為圓形
- 3.想將擊球點角度提高,所以將擊球桿的擊球面設計為平面



主要問題:

- 1.擊球時,球無法順利擊飛

改善方法:

- 1.將擊球桿的擊球面改為曲型

2.4.2

設計緣由:為了讓球順利擊飛,所以將擊球桿的擊球面改為曲型

影片-1:<https://www.youtube.com/embed/C-K8pxvnJJE>

影片-2:<https://www.youtube.com/embed/w2pv5GIT988>

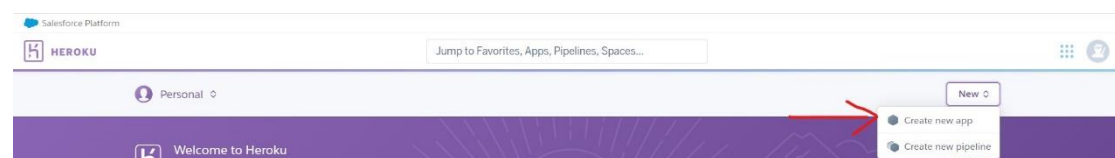
Heroku 網站建立

stage2-ag12-heroku 網站

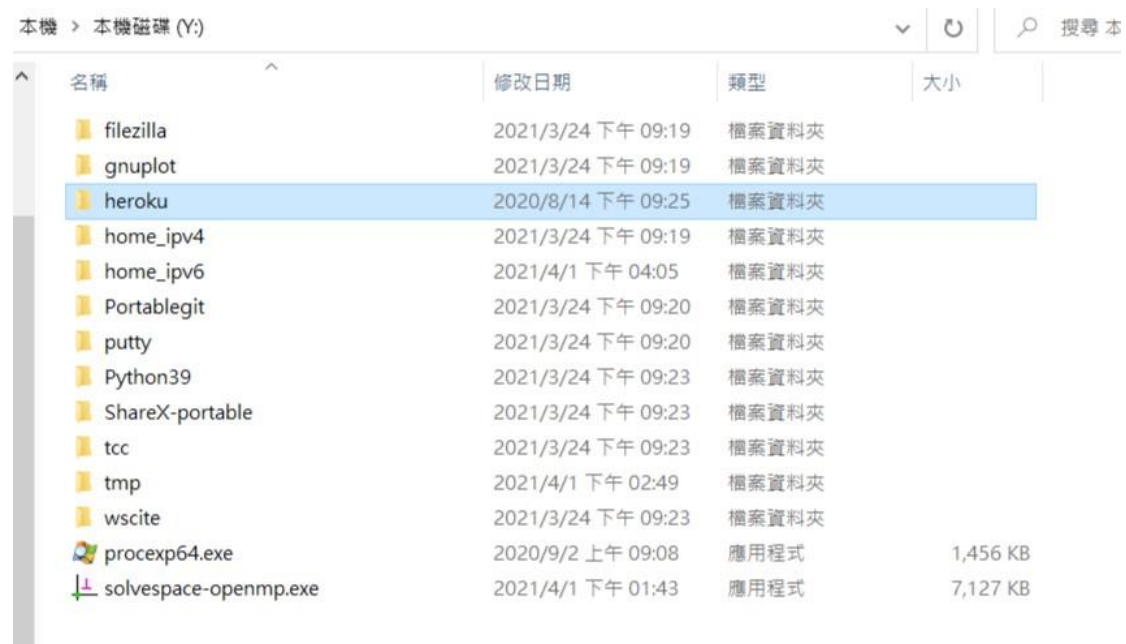
<https://stage2-ag12.herokuapp.com/content/index.html>

1.開通 Heroku 帳號(用學校 email 建立)

2.建立一個 Herokuapp



3.從<http://a.kmol.info:88/heroku.7z>下載 Heroku cli 程式檔案, 解開至隨身碟 Y:



4.修改隨身系統的啟動批次檔案 start.bat:

```

1 start_ipv6.bat
|@echo off
set Disk=y
subst %Disk%: "data"

%Disk%:

set HomePath=%Disk%:\home_ipv6
set HomeDrive=%Disk%:\home_ipv6
set Home=%Disk%:\home_ipv6
set USERPROFILE=%Disk%:\home_ipv6

set GIT_HOME=%Disk%:\portablegit\bin\
set GIT_SSH=%Disk%:\putty\plink.exe
REM 將系統 Python 程式的 io 設為 utf-8
set PYTHONIOENCODING="utf-8"

set PYTHONPATH=%Disk%:\Python39\DLLs;%Disk%:\Python39\Lib;%Disk%
:\Python39\Lib\site-packages;
set PYTHONHOME=%Disk%:\Python39

REM 設定跟 Python 有關的命令搜尋路徑
set path_python=%Disk%:\Python39;%Disk%:\Python39\Scripts;
set path_portablegit=%Disk%:\Portablegit\bin;
set path_tcc=%Disk%:\tcc;
set path_gnuplot=%Disk%:\gnuplot\bin;
set path_Heroku=%Disk%:\heroku\bin;

path=%Disk%:;%path_python%;%path_portablegit%;%path_tcc%;
%path_gnuplot%;%path_Heroku%;%path%;
start /MIN cmd.exe
start /MIN cmd.exe
start /MIN cmd.exe
start /MIN cmd.exe

start /MIN %Disk%:\wScite\SciTE.exe
start /MIN %Disk%:\wScite\SciTE.exe

Exit

```

5.測試是否能夠在修改後的命令列中執行 heroku.exe, 輸入 heroku version 若回應所使用的版本表示上述路徑設定已經成功



```

C:\windows\system32\cmd.exe
Microsoft Windows [版本 10.0.18363.1440]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。
Y:\>heroku version
heroku/7.42.6 win32-x64 node-v12.16.2

```

6.有 proxy 的話，需要在 start_ipv6.bat 中加入:

1. REM for heroku login
2. set HTTP_PROXY=http://[2001:288:6004:17::53]:3128
3. set HTTPS_PROXY=http://[2001:288:6004:17::53]:3128

7. 針對利用 Heroku 同步 Github Pages 上的網站內容

```
Y:\>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/dea27aca-c2c7-4146-aeal-995c2cbeb029?requestor=SFMyNTY.g
2gDbQAAAawMTAuMzAuNjQuMjduBgDU_30MeAFiAAFRgA.vNvifJhnBL6AW4a4N1loupC3VHFwdoYoFL--7MtIik8
Logging in... done
Logged in as 40823125@gm.nfu.edu.tw
```

8.

以 interactive 模式登入 Heroku, 完成後產生 y:\home_ipv6_netrc

1. heroku login -i

針對已經登入的 Heroku 帳號下的 app, 設為此倉儲的 remote 同步倉儲,
且在倉儲中會自建名稱爲 heroku 的 remote site.

2. heroku git:remote -a 名稱

```
C:\windows\system32\cmd.exe
Microsoft Windows [版本 10.0.18363.1440]
(c) 2019 Microsoft Corporation. 著作權所有，並保留一切權利。

Y:\>heroku version
heroku/7.42.6 win32-x64 node-v12.16.2

Y:\>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/dea27aca-c2c7-4146-aeal-995c2cbeb029?requestor=SFMyNTY.g
2gDbQAAAawMTAuMzAuNjQuMjduBgDU_30MeAFiAAFRgA.vNvifJhnBL6AW4a4N1loupC3VHFwdoYoFL--7MtIik8
Logging in... done
Logged in as 40823125@gm.nfu.edu.tw

Y:\>cd tmp\stage2-agl2

Y:\tmp\stage2-agl2>heroku git:remote -a stage2-agl2
set git remote heroku to https://git.heroku.com/stage2-agl2.git

Y:\tmp\stage2-agl2>git push heroku
Enumerating objects: 426, done.
Counting objects: 100% (426/426), done.
Delta compression using up to 8 threads
Compressing objects: 100% (263/263), done.
Writing objects: 100% (426/426), 11.14 MiB | 155.00 KiB/s, done.
Total 426 (delta 161), reused 396 (delta 143), pack-reused 0
remote: Git submodules detected, installing:
remote: Submodule 'cmsimde' (https://github.com/mdecourse/cmsimde.git) registered for path 'cmsimde'
remote: Cloning into 'cmsimde'...
remote: Submodule path 'cmsimde': checked out '9c9d8e91a996a175ce95db1929cc9afead0b76fd'
remote:
```

協同錯誤

在協同時因為組員們的倉儲版本不同導致無法推送

```
Y:\tmp\stage2-ag12>git push 40823153
To github.com:40823153/stage2-ag12.git
! [rejected]        main -> main (non-fast-forward)
error: failed to push some refs to 'github.com:40823153/stage2-ag12.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. Integrate the remote changes (e.g.
hint: 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

最後我們只好將版本不同的倉儲刪掉重新協同

總結:

這次 stage2 分組時,我們學習到了如何跟同學有效的進行設計上的討論,並且解決設計上的問題。

而且我們發現與 stage1 不同 4 人進行協同時,容易有版本錯誤的問題要解決。