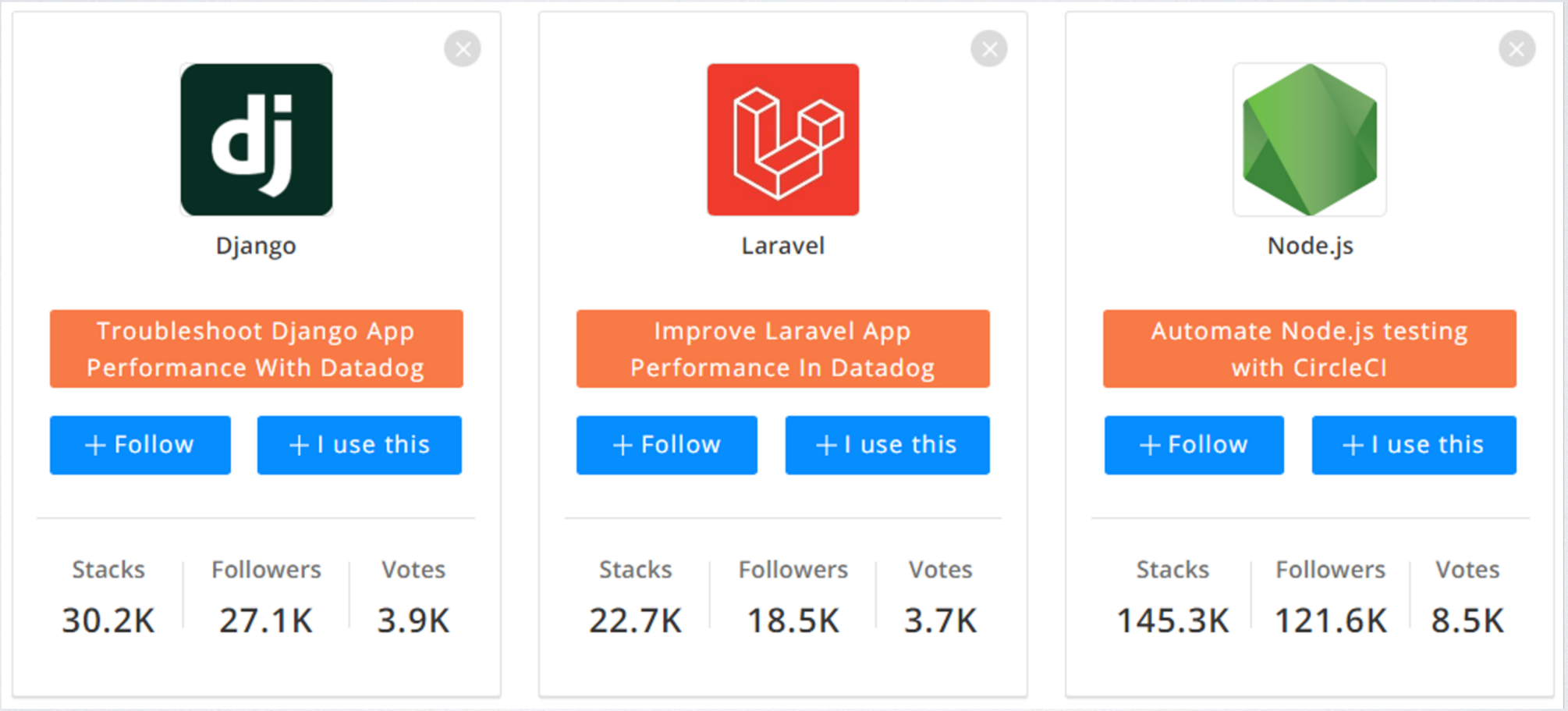
# VUE

高莉淇

# FRONT-END / BACK-END

# BACK-END FRAMEWORK

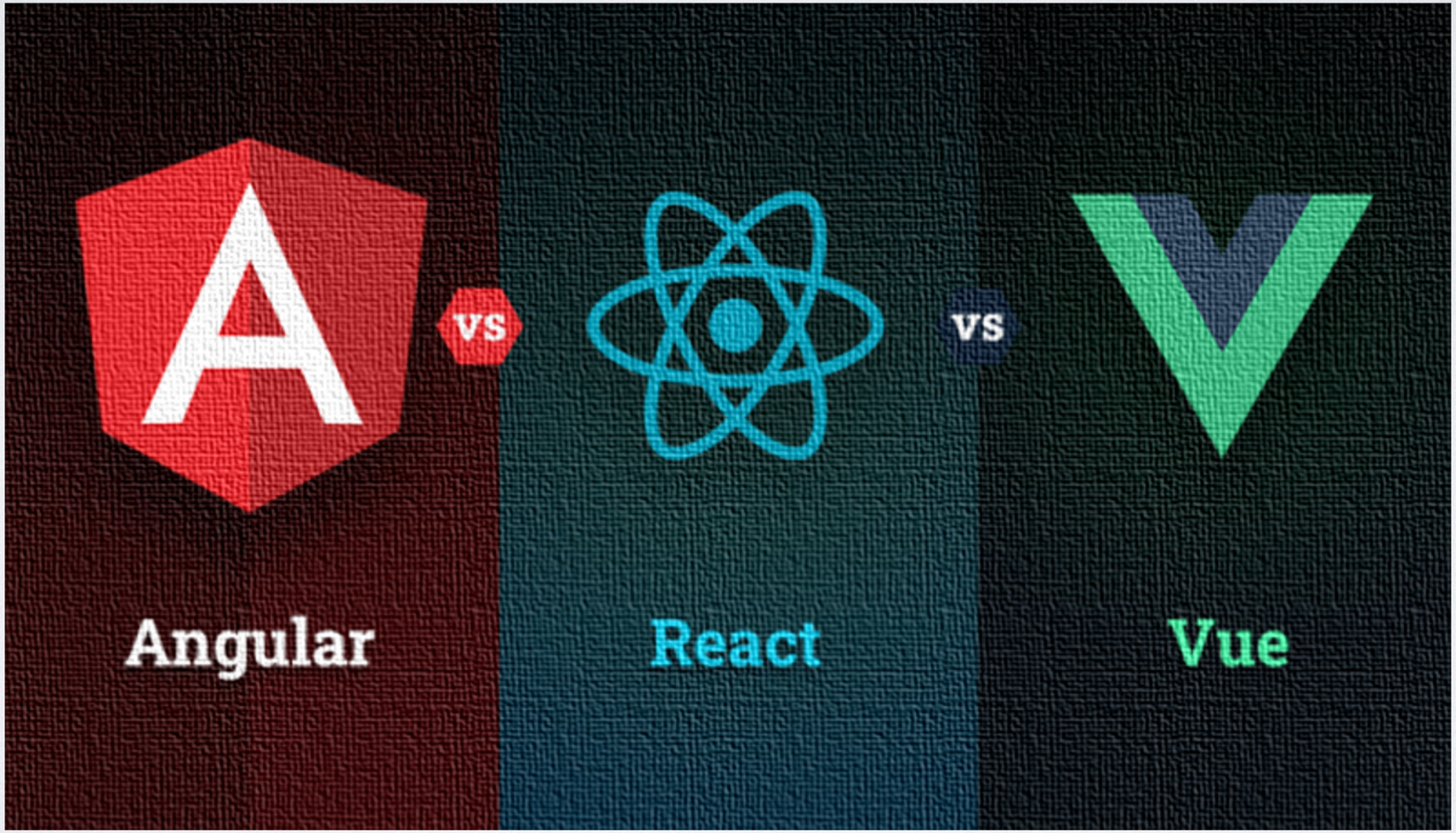


# BACK-END FRAMEWORK

| 框架名稱 | 基於語言 | 相關搭配 | 備註 | 總結 |
|---|---|---|---|---|
| Node Express JS | JS | NodeJS、NPM | | 很新，很紅，社群大 |
| Laravel | PHP | | | 最老，歷久不衰 |
| Django | Python | | | 有一席之地 |
| Flask | Python | Jinja2 (模板引擎) | 輕量化 | 適合快速開發 |
| FastAPI | Python | | 輕量化 | 很新，性能高，社區不完整 |

# BEFORE FRONT-END FRAMEWORK

# FRONT-END FRAMEWORK



# FRONT-END FRAMEWORK

| 框架名稱 | 基於語言 | 開發者 | 著名開發案例 | 總結 |
|---|---|---|---|---|
| Angular | Typescript | Google | Wix | 成熟，學習成本高 |
| React | JS | Facebook | Uber | 相當不錯 |
| Vue | JS | 前Google工程師 | Gitlab | 較新，易學，備受矚目 |

| JavaScript套件 | 總結 |
|---|---|
| jQuery | 完整，易學，不易維護 |

# FLASK(JINJA) & VUE3.0

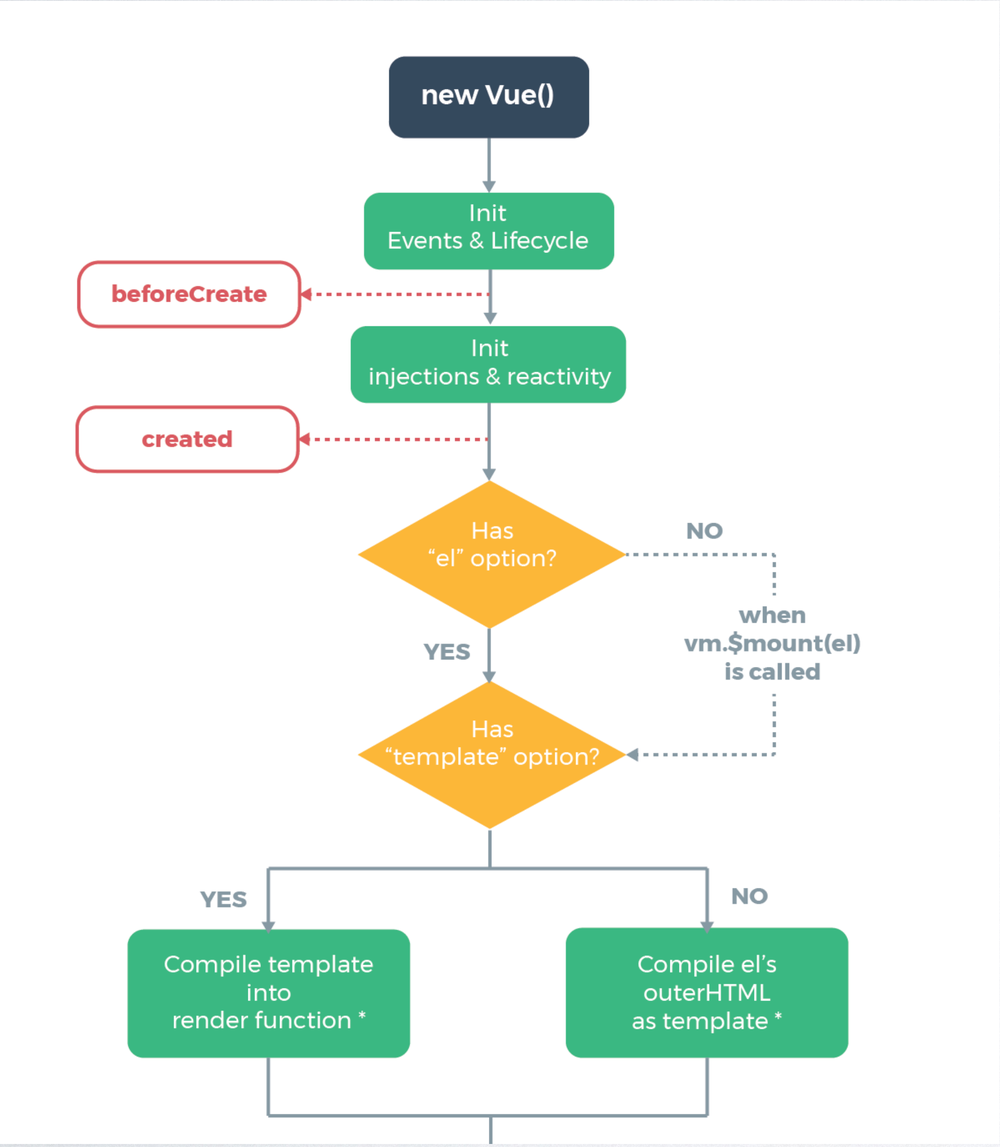
# CDN

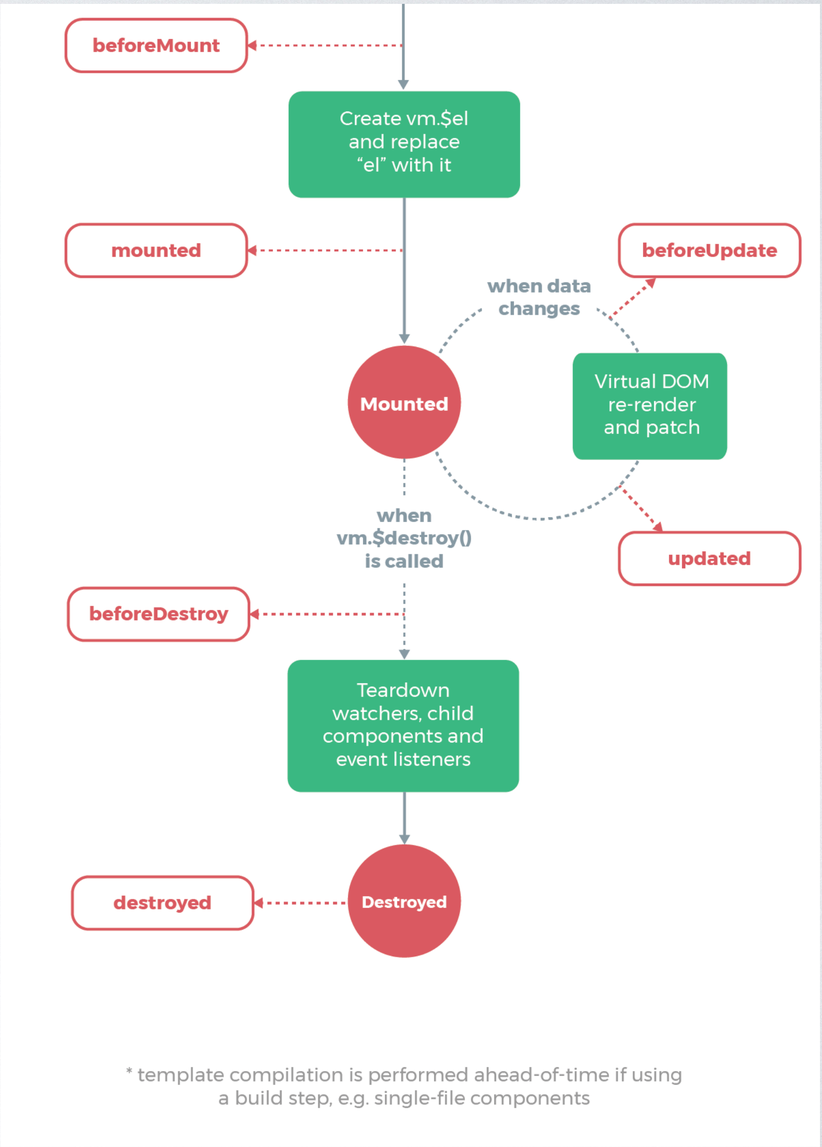
- 在 flask 的框架下，只能使用 CDN 來引用 js

- **cdnjs**
  - 選擇穩定版本（沒有alpha 或 beta）
  - 複製 URL or Script Tag

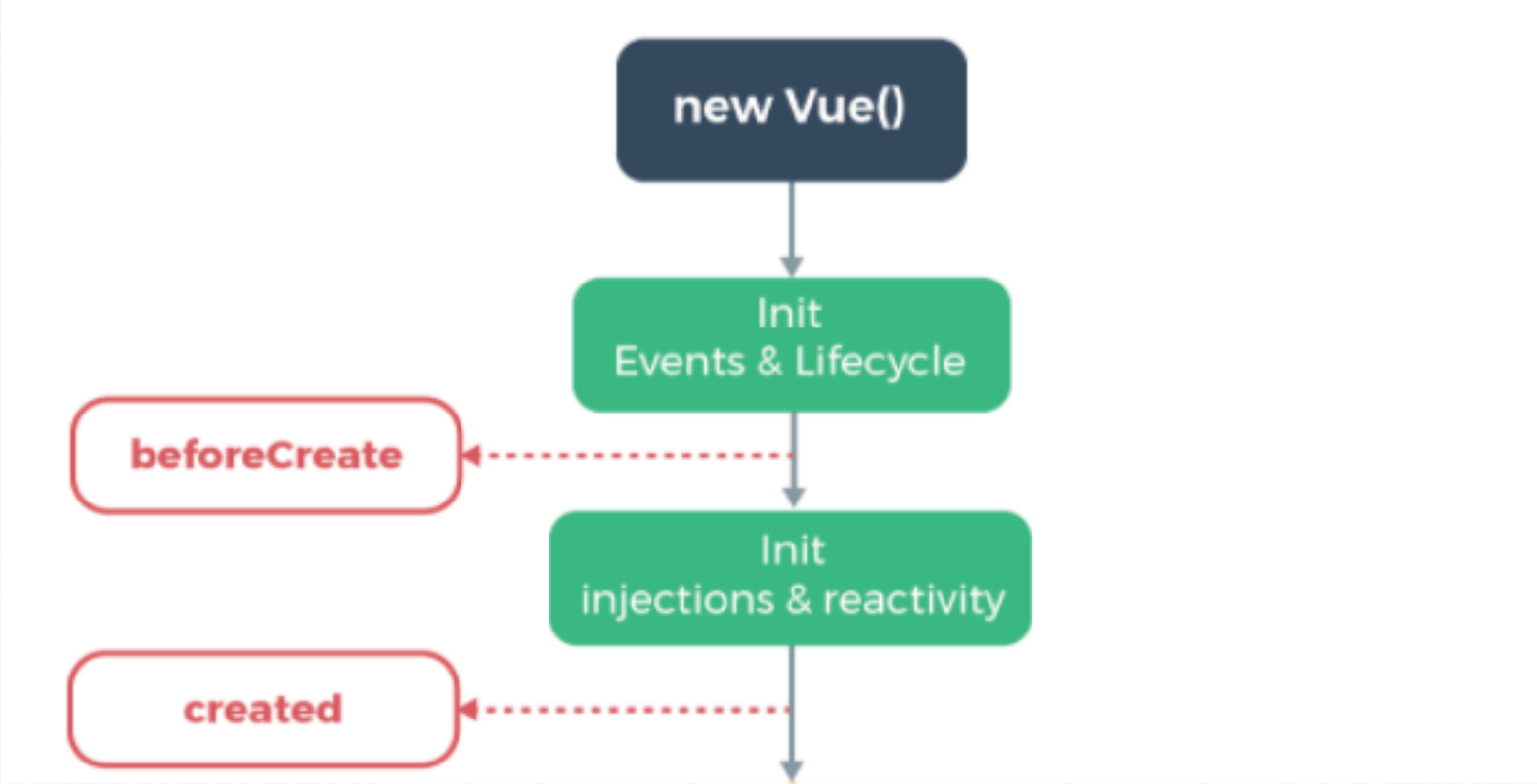
# LIFE CYCLE

# LIFE CYCLE

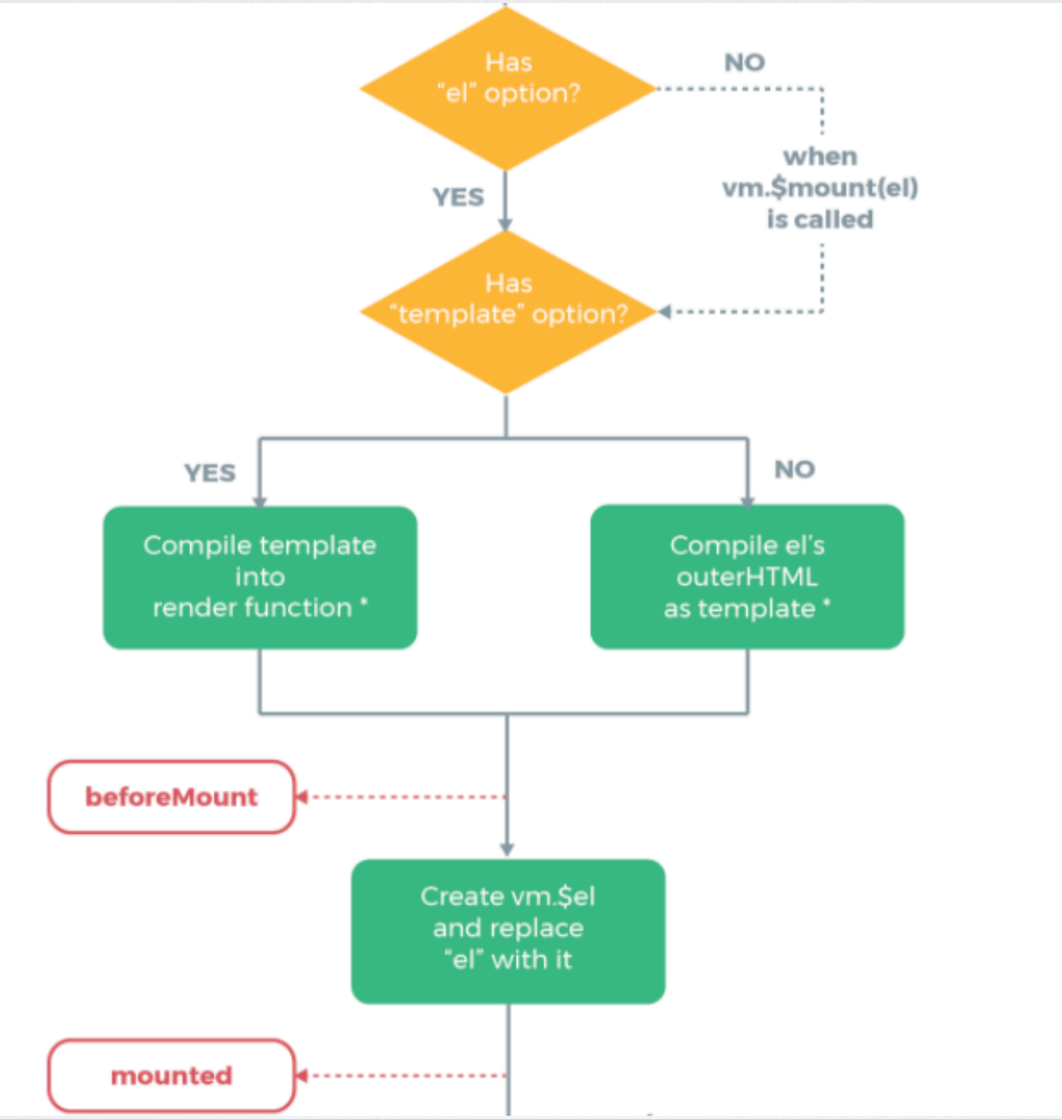

new Vue()
Init
Events & Lifecycle
beforeCreate
Init
injections & reactivity
created
Has
“el” option?
NO
when
vm.$mount(el)
is called
YES
Has
“template” option?
YES
NO
Compile template
into
render function *
Compile el’s
outerHTML
as template *


beforeMount
Create vm.$el
and replace
“el” with it
mounted
beforeUpdate
when data
changes
Mounted
Virtual DOM
re-render
and patch
when
vm.$destroy()
is called
updated
beforeDestroy
Teardown
watchers, child
components and
event listeners
destroyed
Destroyed
* template compilation is performed ahead-of-time if using
a build step, e.g. single-file components

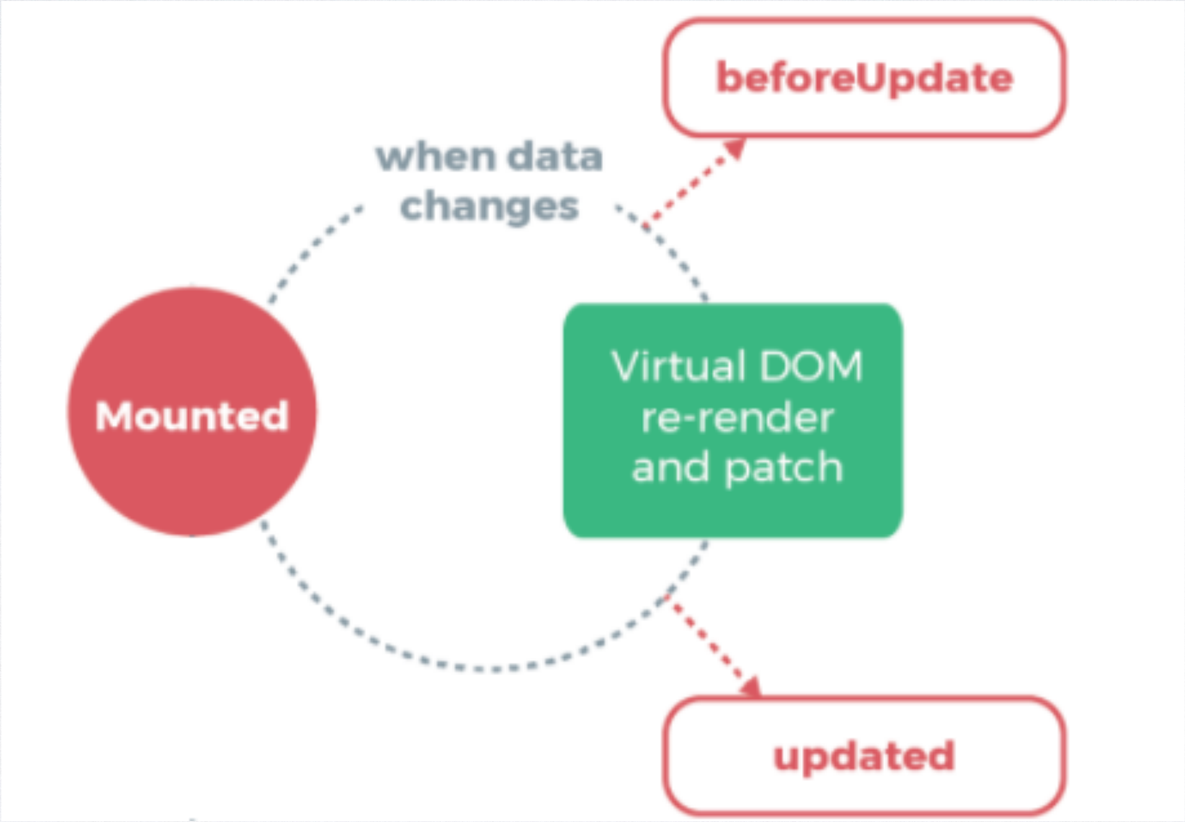
# CREATION：元件初始化階段



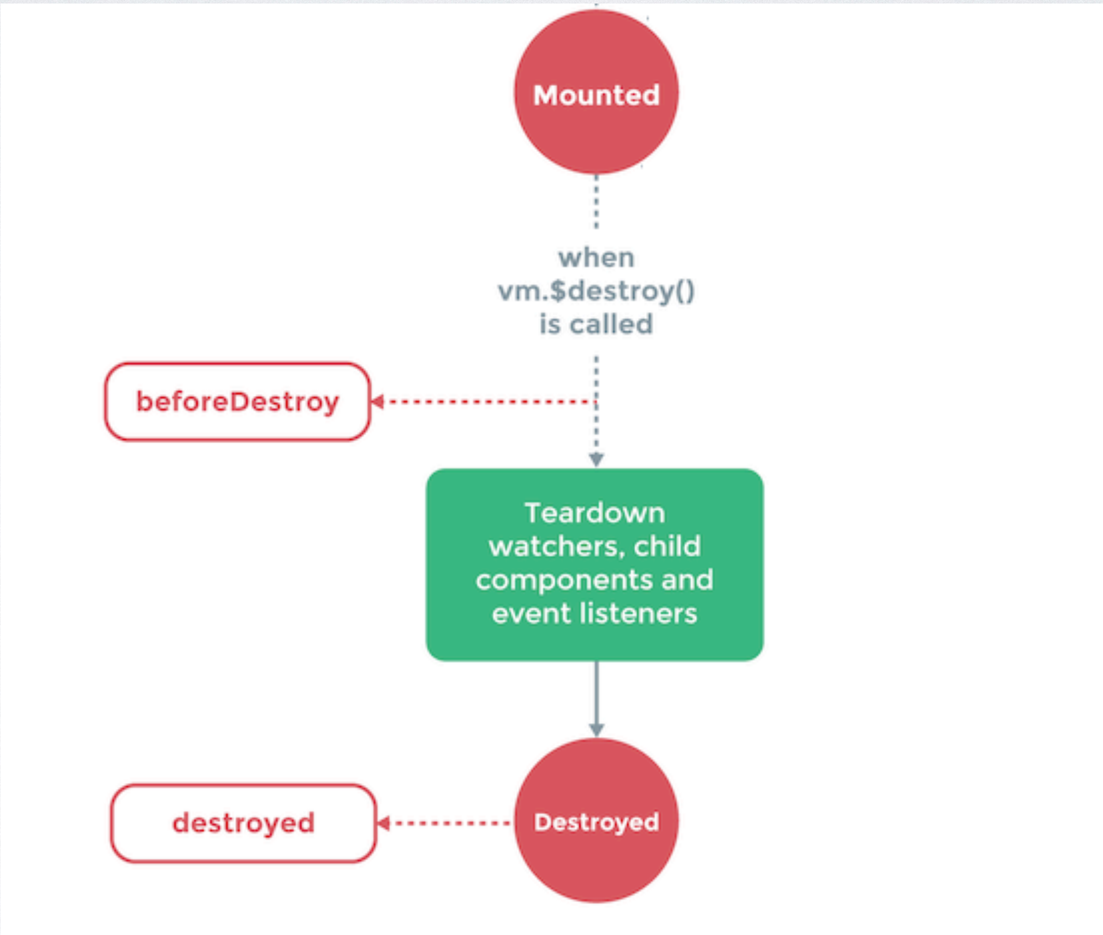
# MOUNTING：掛載階段



# UPDATING：資料更新階段，重新渲染畫面



# DESTRUCTION: 元件銷毀階段



# SHOW VARIABLE

# SHOW VARIABLE

- 原本
  - Jinja 以 {{ }} 代入變數
  - Vue 以 {{ }} 代入變數
- 更改
  - Jinja 以 % % 代入變數
  - Vue 以 {{ }} 代入變數

```
from flask import Flask, render_template, request

class VueFlask(Flask):
    jinja_options = Flask.jinja_options.copy()
    # Default is "{{ }}", changing to "%% %%"
    jinja_options.update(dict(variable_start_string="%%",
                              variable_end_string="%%"))
app=VueFlask(__name__)
```

# SHOW VARIABLE

- 搭配文件：1_jinja_vue.html
- 練習顯示你的學號（用jinja） 和 你的名字（用vue）

# CONTROL VARIABLE

# CONTROL VARIABLE

- 使用 {{ }} 顯示
- 使用 v-bind 操作屬性
- 使用 v-model 取得使用者輸入

- 使用 v-on:click="FUNC"，觸發按鈕
- 使用 v-on:keyup.enter="FUNC"，觸發ENTER

- FUNC 要寫在 methods 下
- 監聽要寫在 watch 下

# CONTROL VARIABLE

- 搭配文件：2_vue_variable.html

- 練習顯示請使用VUE 取得 Bootstrap Dropdowns 的值，並顯示出來
- Dropdowns

# FLOW

# FLOW

- 使用 v-if v-else-if v-else 控制顯示
- 使用 v-for 顯示大量資料

# FLOW

- 搭配文件：3_vue_flow.html

- 練習產生6個按鈕，按鈕名稱帶有編號
- 奇數號按鈕是紅的，偶數號按鈕是綠的

# AXIOS

# AXIOS

- Axios是Vue官方推薦的非同步工具
- 使用 get、post、 push 等Restful (Representational State Transfer: 表現層狀態轉換) 規範來取得第三方資料
- .then 表示取得資料後的處理動作。
- 相當於 Jquery 裡的 ajax

# AXIOS

- GET

```python
@app.route("/test/get", methods = ["GET"])
def test_get():
    yorn = request.args.get("yorn")
    print(yorn)
    if yorn == "y":
        result = {
            "status": "success"
        }
    else:
        result = {
            "status": "fail"
        }
    return result
```

app.py

```javascript
axiosget(val) {
    axios
        .get(`/test/get?yorn=${val}`)
        .then((response) => {
            console.log("response data:", response.data);
            if (response.data.status == "success")
                this.getresult = "get success";
            else
                this.getresult = "get fail";
        })
        .catch((error) => {
            alert(error);
        })
        .finally(() => {
            console.log("finally");
        });
},
```

4_vue_axios.html

# AXIOS

- POST

```python
@app.route("/test/post", methods = ["POST"])
def test_post():
    data = request.get_json()
    yorn = data["yorn"]
    print(yorn)
    if yorn:
        result = {
            "status": "success"
        }
    else:
        result = {
            "status": "fail"
        }
    return result
```

app.py

```javascript
axiospost(val) {
    axios
        .post('/test/post', {
            yorn: val
        })
        .then((response) => {
            console.log("response data:", response.data);
            if (response.data.status == "success")
                this.postresult = "post success";
            else
                this.postresult = "post fail";
        })
        .catch((error) => {
            alert(error);
        })
        .finally(() => {
            console.log("finally");
        });
},
```

4_vue_axios.html

# AXIOS

- 搭配文件：4_vue_axios.html

- 練習
- 請去下列網址取得Bitcoin，並分別顯示「美元、英鎊、歐元」的當前價格
- https://api.coindesk.com/v1/bpi/currentprice.json

# HOMEWORK

# HOMEWORK 4

1. 自己想一個題目，建立一個網站
2. 修改現有網站：

   ① 合成網站（http://140.116.245.146:30011/taiwanese_synthesis）

   - 問題留言區 - Ajax, bootstrap

- 八月底報告給老師聽