FALL 2015

Coventry University

Faculty of Engineering and Computing

## Module Code 321COM

## Module Title Rapid Application Development

Instructions to candidates

Answer 4 out of 5 Questions

Time allowed: 3 Hours 00 minutes

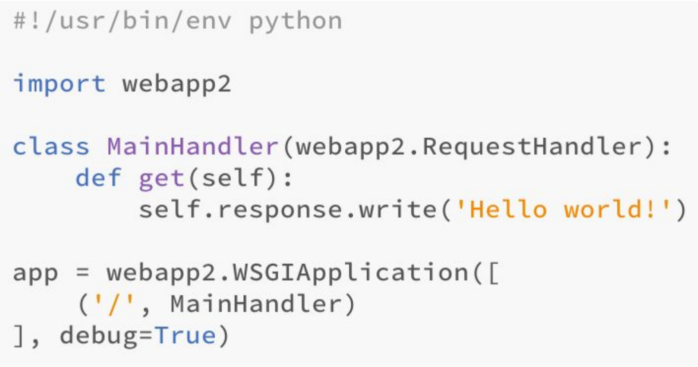
For this examination you will be supplied with the following:

**IMPORTANT:** You may take this question paper away at the end of the examination. Please keep it in a safe place for future reference.

continued…

<table>
<tr><td colspan="3">Question 1</td></tr>
<tr><td>a)</td><td>To reduce security risks, GAE applications run in ‘sandboxes’ which prevents one application interactive with others. What are the FOUR imposed restrictions.

- Applicants can’t write to the filesystem (must use a datastore)
- Applicants can only communicate using ports 80 and 443
- Applicants can’t take more than a few  seconds to respond to requests (can’t tie up system resources)
- Applicants can’t make system calls
</td><td>8 marks</td></tr>
<tr><td>b)</td><td>A POST request with a form input named “contact” and path “/” is sent to your application. Rewrite the following code to response to the POST request and print the content of “contact” on client’s browser.



```
#!/usr/bin/env python

import webapp2

class MainHandler(webapp2.RequestHandler):
    def get(self):
        self.response.write('Hello world!')

app = webapp2.WSGIApplication([
    ('/', MainHandler)
], debug=True)
```

```
def post(self):
    content = self.request.get(‘content’)
    self.response.write(content)
    self.redirect(‘/’)
```
</td><td>9 marks</td></tr>
<tr><td>c)</td><td>It is a very common to use callback pattern to handle request from client in GAE webapp. The basic idea is that we hand over the main responsibility for handling something to a framework. Explain the callback pattern in action of GAE web application.<br>The incoming HTTP request arrives to our main program.<br>Instead of handling the request directly,<br>we simply set up the framework and tell it under what conditions (urls that match /.*) and where (MainHandler) to call us back when it needs some “assistance” from us.

Then the framework starts up and looks at the HTTP request, figuring out which kind of request it is - parsing</td><td>8 marks</td></tr>
</table>

continued…

| | | |
|---|---|---|
| | all of the data, converting file input if necessary - and then calls out MainHandler - using either the get() or post() method as appropriate. | |

continued…

Question 2

a) Compare and contrast the POST and GET requests in HTTP protocol in terms of their structures, usages, size limit, the ability to cache and the appropriateness to change the server.

10 marks

| GET | POST |
|---|---|
| ● Parameters in URL | ● Parameters in body |
| ● Used for fetching documents | ● Used for updating data |
| ● Maximum URL length | ● No maximum length |
| ● OK to cache | ● Not OK to cache |
| ● shouldn’t change the server | ● OK to change the server |

b) Given that the following JSON code:

15 marks

```
//JSON, from http://website.com/data.json
{
  "items" : [
     {
         "snippet" {
              "title":"Places to visit in Coventy."
          }
    },
    {
         "snippet" {
              "title":"Hello World"
          }
    }
  }
}
```

Rewrite the script to download, parse and print every “title” of each items according to the code below.

```
import urllib2
import json
url = http://website.com/data.json


class MainHandler(webapp2.RequestHandler):
    def get(self):
getJSON = urllib2.urllib.urlopen(url).read()
loadJSON = json.loads(getJSON)

for result in loadJSON[‘item’]:
    self.response.write( result[”snippet”][“title”])
```

continued…

continued…

| Question 3 | | |
|---|---|---|
| a) | Discuss the differences between strong consistency and eventual consistency.<br>討論強一致性和最終一致性之間的差異。<br>Strong<br>data is always consistent, no matter what happens<br>Eventual<br>eventually data is consistent, but there is some time when data is not<br>**Strong consistency:**<br>1. Makes it very hard to ensure scalability without scarifying<br>2. performance<br>3. We can scale by replicating or sharding on different machines<br>4. Then to ensure strong consistency the write operations will be very slow due to locking<br>5. Join operation will be very slow too<br>Strong consistency for a single row,<br>eventual consistency for multi-row levels<br>強一致性：<br>很難確保可擴展性而不會影響性能<br>我們可以通過在不同的機器上複製或分片進行擴展<br>然後為了確保強大的一致性，由於鎖定，寫入操作將非常緩慢<br>加入操作也會非常慢<br>單行強一致性，<br>最終的多行級別的一致性 | 4 marks |
| b) | Rational database management system (RDBMS) always maintains a strong consistency. Explain why the strong consistency make RDMS not as good as bigtable in handling very large size data.<br>Rational數據庫管理系統（RDBMS）始終保持強大的一致性。解釋為什麼強大的一致性使RDMS在處理超大規模數據方面不如bigtable好。<br>Bigtable is not a database<br>Bigtable does not support queries | 8 marks |

continued…

Because Bigtable is a sparse, distributed, persistent multidimensional map

c)

Given the following ndb.Model subclass:

```
from google.appengine.ext import ndb

class Account(ndb.Model):
        name = ndb.String Property()
        userid = ndb.IntegerProperty()
        googleid = ndb.UserProperty()
        date = ndb.DateProperty()
        dob = ndb.DateProperty(auto_now_add=  True)
```

Based on the given class, write a code fragment to create a new Account entry and store it into the datastore.

```
class create Account(webapp2.RequestHandler):
        def post(self):
             name = “CHAN Tai Man”
             userid = 12345
             googleid = users.get_current_user()
             date = datetime.date(2000,1,31)
             storeAccountInformation = Account(
                                     name = name,
                                     userid = userid,
                                     googleid=googleid,
                                     date=date
             )
        storeAccountInformation.put()

class showAccount(BaseHandler):
         def get(self):
              results = Account.query().fetch()
              for result in results
                   self.response.write(result)
```

9 marks

d)

ACID is a set of properties that guarantee that database transactions are process reliably. What do the “A”, “C”, “I” and “D” mean?

ACID是一組保證數據庫事務處理可靠的屬性。 “A”, “C”, “I”和“D”是什麼意思？

- Atomicity

4 marks

continued…

| | | |
|---|---|---|
| | ● Consistency<br>● Isolation<br>● Durability<br>● 原子性<br>● 一致性<br>● 隔離<br>● 耐久力 | |

continued…

| Question 4 | | |
|---|---|---|
| a) | Waterfall model is a traditional methodology used in software development processes.<br>Why is waterfall model not able to create and respond to change in a turbulent business environment?<br>瀑布模型是軟件開發過程中使用的傳統方法。<br>為什麼瀑布模型無法在動蕩的商業環境中創建和響應變化？<br>1. Difficult to accommodate change after the process is underway<br>2. One phase has to be completed before the next<br>3. Appropriate only for projects with well understood and stable requirements, very large projects<br>1. 這個過程正在進行之後，很難適應變化<br>2. 一個階段必須在下一個階段之前完成<br>3. 只適用於需求清晰，穩定的項目，非常大的項目<br>什麼樣的項目適合使用這種模式？<br>What kind of project is suitable to use such model?<br>agile methodologies.<br>Work is implemented in stages (iterations), and only enough planning is carried out to complete the next iteration<br>敏捷方法。<br>工作分階段（迭代）實施，只有足夠的計劃才能完成下一次迭代 | 9 marks |
| b) | Name any TWO agile methodologies.<br>命名任何兩個敏捷方法。<br>1. Scrum - [ 混戰 ]<br>2. Crystal Methods - [ 水晶方法 ]<br>3. Lean Development (LD) - [ 精益發展（LD） ]<br>4. Extreme Programming (XP) - [ 極限編程（XP） ]<br>5. Feature-Driven Development (FDD) - [ 功能驅動開發（FDD） ]<br>6. Adaptive Software Development (ASD) - [ 自適應軟件開發（ASD） ]<br>7. Dynamic Systems Development Method (DSDM) - [ 動態系統開發方法（DSDM） ] | 6 marks |
| c) | Discuss the way to manage the following rick factors in agile methodology:<br>1. Schedule slips<br>2. Project cancelled<br>3. Business misunderstood<br>4. Defect rate<br>5. False feature rich | 2 marks<br>2 marks |

continued…

| | | |
|---|---|---|
| | Schedule slips - Short release cycles<br>Project cancelled - Smallest release that makes sense<br>Business misunderstood - Make the customer part of the team<br>Defect rate - Testing by programmers and customers<br>False feature rich - Address only the highest priority tasks<br><br>System goes sour - Maintain a suite of tests<br>Business changes - Short release cycles<br><br>supported by google translate<br>討論如何管理敏捷方法中的以下rick因素：<br>1. 附表滑倒<br>2. 項目取消<br>3. 業務被誤解了<br>4. 缺陷率<br>5. 功能豐富<br>supported by google translate | 2 marks<br><br>2 marks<br><br>2 marks |

| Question 5 | | |
|---|---|---|
| a) | Name any 3 typical web application security breaches.<br>命名任何3個典型的Web應用程序安全漏洞。<br>1. Validate user input<br>2. Set correct database permissions<br>3. Use stored procedures | 6 marks |
| b) | Provided that the following SQL query:<br>SELECT * From Account WHERE username = ‘$user’ AND password = ‘$pass’<br><br>Assumed that there is no any security implementation in this web application, how would you delete all user accounts based on above SQL statement using SQL injection?<br><br>假設以下SQL查詢：<br>選擇*從帳戶WHERE用戶名='$用戶'和密碼='$傳遞'<br><br>假設在這個Web應用程序中沒有任何安全實現，<br>那麼如何使用SQL注入基於上述SQL語句刪除所有用戶帳戶？ | 8 marks |

continued…

| | | |
|---|---|---|
| | SELECT * From Account WHERE username = ‘’;<br>DELETE * FROM `Account` WHERE “abc or user==user”/;’’ AND password = ‘’; | |
| c) | What is cross-site script (XSS)? And what are the impacts of XSS attack?<br>Cross-site script refer to Attacker sends raw data to a user’s browser<br>This data could come from:<br>1. a database<br>2. a form they are filling in<br>3. added directly into the client<br>Impact XSS attack<br>1. steal a user’s session<br>2. steal sensitive data<br>3. change the content of a web page<br>4. redirect users to a phishing or malware site<br>5. install a proxy to observe and direct the user | 7 marks |
| d) | Suggest any two programming approaches, which are able to defend against cross-site scripting.<br>建議任何兩種編程方法，可以防範跨站腳本。<br>Recommendations:<br>1. Eliminate Flaw - [ 消除缺陷 ]<br>2. Defend Against the Flaw - [ 防禦缺陷 ]<br>3. Don’t include user supplied input in the output page - [ 不要在輸出頁面中包含用戶提供的輸入 ] | 4 marks |

Final page of exam paper

continued…

# Tips

- GAE
    - GAE characteristics advantages (security)
    - run sandbox benefit
    - limit port
    - execution timeout
    - can not read local file
    - can not run system call
    - GAE coding

- GAE Coding
    - understand project structure (file? / python class?)
    - handler (coding)
    - configuration file

- Datastore
    - rdbms vs datastore?
        - consistency (strong? Eventual?)
        - coding create ndb mode insert,query

- Web API
    - POST vs GET
    - json vs xml (markup format)
    - coding -urllib2 download, read json into dictionary / array

- PM (Scrum / Agile)
    - definition, waterfall vs scrum
    - advantages of scrum -pm
      (i.e., how to manage risk: schedule slip / project cancelled?)
    - user story - how to write? (template), story validation?
    - conditions of satisfaction > acceptance test
    - scrum vs extreme programming ( or other methodology)

- Security
    - Inject / XSS
    - How injection works with example? what harmful thing can injection do?
      (about SQL)
    - How does XSS work with example? what harmful thing can XSS do?
      (about fake website)

**oooOooo**

continued…