

به نام خدا

آزمون پایان ترم

نیم سال دوم ۱۴۰۱-۱۴۰۲

۴ تیر ۱۴۰۲

سوالات تستی (هر کدام ۱۰ نمره)

1. منظور از failure در آزمون نرم افزار چیست؟
 - a. حضور یک fault در نرم افزار
 - b. یک error که در طی اجرای نرم افزار رخ می دهد
 - c. یک اشکال در عملکرد یا انحراف از رفتار مورد انتظار نرم افزار
 - d. یک defect که در طی فاز تست نرم افزار کشف شده
2. یکی از ویژگی های آزمون خوب برای یک نرم افزار، Best of Breed بودن است. منظور از این ویژگی برای یک آزمون چیست؟
 - a. بین یک دسته از آزمون ها که هدف، منابع زمانی و سخت افزاری مشابهی دارند، آزمونی انتخاب شود که بتواند دسته ای کامل از خطاها را شناسایی کند.
 - b. یک آزمون نباید هدفی یکسان با آزمونی دیگر داشته باشد.
 - c. آزمون باید با احتمال خوبی بتواند خطاهای آن بخش از نرم افزار را که برای آن نوشته شده است، پیدا کند.
 - d. آزمون باید به صورت تخصصی برای یک بخش کوچک از برنامه نوشته شود. تست نباید بزرگ باشد و بخواهد کار چندین آزمون را در خودش انجام دهد و بخش بزرگی از ویژگی های برنامه را پوشش دهد.
3. درباره انواع روش های آزمون یک نرم افزار، کدام گزینه نادرست است؟
 - a. روش آزمون جامع¹، تمامی مسیرهای ممکن در یک برنامه را پیمایش می کند تا بتواند صحت برنامه را در تمامی سناریوهای ممکن بیازماید. با استفاده از پردازنده های سریع مخصوص این نوع از آزمون ها می توان این نوع از آزمون را در زمان معقولی انجام داد.

¹ Exhaustive Testing

- b. روش‌های جعبه‌سفید² و جعبه‌سیاه³ در آزمون نرم‌افزار، روش‌هایی مادر برای دیگر روش‌ها هستند که تفاوت آن‌ها در دانستن ساختار داخلی برنامه توسط آزمونگر نرم‌افزار است.
- c. روش آزمون بر پایه مدل⁴، فرض می‌کند که به مدل‌های رفتاری که در مراحل ایجاد نرم‌افزار ایجاد شده‌اند، دسترسی دارد.
- d. روش طراحی آزمون مدل-محور⁵، فاصله‌ی روش‌های آزمون جعبه‌سفید و جعبه‌سیاه را کم‌رنگ می‌کند و ریاضیاتی‌ترین و فنی‌ترین روش آزمون نرم‌افزار است.

3. کدام گزینه در مورد Regression Testing درست است؟

- a. به ما کمک می‌کند تا مطمئن شویم تا قسمت‌هایی از نرم‌افزار که بهشان دست نزده‌ایم، خراب نشده‌اند.
- b. فقط در حین فرآیند User Acceptance Test قابل اجرا است.
- c. همواره به صورت خودکار انجام می‌شود.
- d. پس از commit شدن تغییری روی مخزن کد، Regression Test انجام می‌شود.

5. کدام یک از گزینه‌های زیر، جزء فواید روش افراز فضای ورودی⁶ برای آزمون یک method محسوب نمی‌شود؟

- a. می‌توان این روش را بدون خودکارسازی اعمال کرد.
- b. این روش می‌تواند در مراحل مختلف آزمون از جمله آزمون سیستم⁷، آزمون ادغام⁸ و ... اعمال شود.
- c. برای استفاده از این روش و فهمیدن فضای ورودی، نیاز به دانش حداقلی نسبت به پیاده‌سازی method است.
- d. این روش منطبق با نیاز آزمونگر، می‌تواند تعداد کم یا زیاد آزمون تولید کند.

6. کدام گزینه در مورد Beta Testing صحیح می‌باشد؟

- a. روشی ارزان برای اجرای یک برنامه تضمین کیفیت نرم‌افزار (software quality assurance) می‌باشد.
- b. روشی معقول برای آزمودن نرم‌افزار مستقرشده روی زیرمجموعه‌ای از کاربران نهایی است.
- c. بهترین راه برای دریافت بازخورد متمرکز کاربر برای پشتیبانی از بهبود رابط کاربری نرم‌افزار می‌باشد.
- d. راهی تضمین شده برای به دست آوردن تبلیغات مثبت و شهرت برای محصول می‌باشد.

² White-box Testing

³ Black-box Testing

⁴ Model-based

⁵ Model-driven Test Design

⁶ Input-space Partitioning

⁷ System Testing

⁸ Integration Testing

7. کدام گزینه در مورد Integration Testing که در سطح کوچک انجام می‌شود، صحیح می‌باشد؟

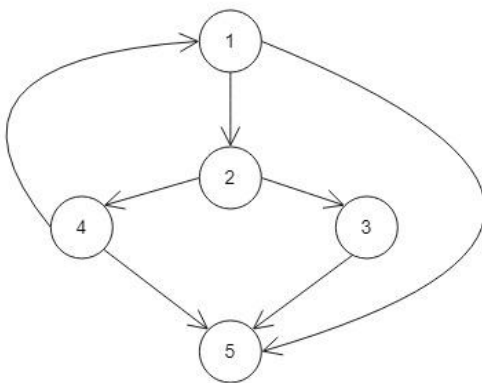
- a. کامپوننت‌های تکی⁹ را تست می‌کند.
- b. فقط از کامپوننت‌های سیستم مستقرشده در محیط کاربر را می‌آزماید.
- c. تعامل بین ماژول‌ها یا زیرسیستم‌های تکی را آزمایش می‌کند.
- d. interface هایی را که هر سیستم در اختیار سیستم‌های دیگر می‌گذارد، آزمایش می‌کند.

8. رشید در حال اجرای یک تست است که با نیاز کاربر برای پر کردن یک فیلد خاص با استفاده از یک Drop Down Box که لیستی از مقادیر دارد، مطابقت دارد. در این زمان کدام یک از انواع تست در حال اجرا شدن است؟

- a. White-box Testing
- b. Black-box Testing
- c. Load Testing
- d. Regression Testing

9. هنگامی که فرآیند Verification and Validation یک نرم‌افزار ایجاد شد...

- a. دیگر هرگز تغییر نمی‌کند.
- b. می‌توان آن را تا زمان شروع قرارداد تغییر داد.
- c. می‌توان آن را تا زمان شروع ایجاد (Development) واقعی نرم‌افزار تغییر داد.
- d. می‌توان آن را در طول چرخه حیات نرم‌افزار تغییر داد.



10. تصویر زیر گراف کنترل حالات یک برنامه‌ی فرضی است

(۱ حالت آغازین و ۵ حالت پایانی است)؛ هر کدام از گزینه‌های زیر دو مسیر را مشخص کرده است. در کدام

گزینه، دو مسیر مستقل از هم وجود دارد؟

- a. (5, 1) و (5, 4, 2, 1)
- b. (5, 3, 2, 1) و (5, 4, 2, 1)
- c. (5, 3, 2, 1) و (5, 4, 2, 1, 4, 2, 1)
- d. (5, 4, 2, 1) و (5, 4, 2, 1, 4, 2, 1)

⁹ Individual

سوالات تشریحی

سوال اول: گراف جریان (۲۰ + ۱۰ + ۱۰ نمره)

```
1  int a, b, c
2  bool isTriangle
3  Output(enter a, b and c integers)
4  Input(a, b, c)
5  Output(side 1 is {a})
6  Output(side 2 is {b})
7  Output(side 3 is {c})
8
9  if (a<b+c) and (b<a+c) and (c<b+a)
10     isTriangle = True
11 else
12     isTriangle = False
13
14 if isTriangle
15     if (a == b) and (b == c)
16         Output("equilateral")
17     else if (a != b) and (a != c) and (b != c)
18         Output("scalene")
19     else
20         Output("isosceles")
21 else
22     Output("not a triangle")
```

با توجه به تکه‌کد زیر به سوالات پاسخ دهید.

الف) گراف جریان مربوط به این تکه‌کد را رسم کنید. برای سادگی، می‌توانید هر خط را معادل یک راس گراف در نظر بگیرید.

ب) یک مجموعه‌ی پایه از مسیرهای خطی مستقل پیدا کنید.

پ) مجموعه‌ای کمینه از موارد آزمون را پیدا کنید که تمام مسیرهای قسمت قبل را پوشش دهد.

سوال دوم: آزمودن متد (۲۰ + ۱۰ + ۱۰ + ۱۰ نمره)

متدی داریم که سه عدد صحیح ورودی می‌گیرد و به ما نوع مثلثی را که می‌توانیم با آن بسازیم، خروجی می‌دهد. در این مسئله فرض کنید ما در عالم تنها سه نوع مثلث مختلف الاضلاع، متساوی‌الساقین و متساوی‌الاضلاع را داریم!

1. با استفاده از یک یا چند خصوصیت دلخواه، ورودی‌های این متد را به دسته‌های مختلف افزار کنید.
2. خصوصیت یا خصوصیت‌هایی را که در بخش قبل استفاده کردید، مبتنی بر عملکرد هستند یا مبتنی بر رابط؟
3. نیازمندی‌های آزمون برای پوشش All Combination به دست بیاورید.
4. کمترین تعداد مورد آزمون را برای پوشش نیازمندی‌های آزمون مرحله قبل، بنویسید.

سوال سوم: گزاره‌های منطقی (۴ + ۱۸ + ۸ + ۱۲ نمره)

با توجه به predicate های داده شده در هر قسمت به سوالات آن پاسخ دهید:

$$P_1 = (a \wedge b) \vee (a \wedge \neg b)$$

1. cluse های P_1 مشخص کنید.

2. هر clause چگونه عبارت P_1 را تعیین می‌کند؟

3. درباره‌ی بخش قبل توضیح دهید که جواب به دست آمده درباره دو متغیر a و b چه معنایی دارند؟

$$P_2 = \forall a \in \mathbb{Z}, \exists b \in \{-1, 1\} \mid (f(a) \wedge (b == 1)) \vee (\neg f(a) \wedge (b == -1))$$

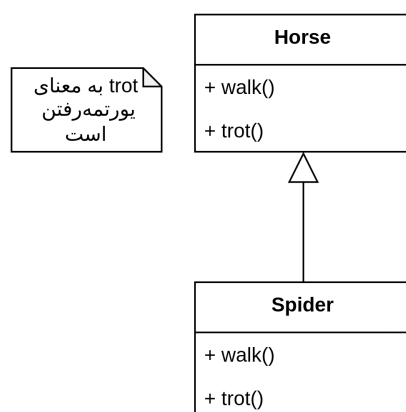
$$f(x): (x \bmod 2) == 0$$

4. P_2 چه عملیاتی را نشان می‌دهد؟

5. موارد آزمون برای تحقق نیازمندی‌های آزمون حاصل از Clause Coverage و Predicate Coverage در P_2 را بنویسید.

سوال چهارم: اصول SOLID (هر مورد ۲۵ نمره)

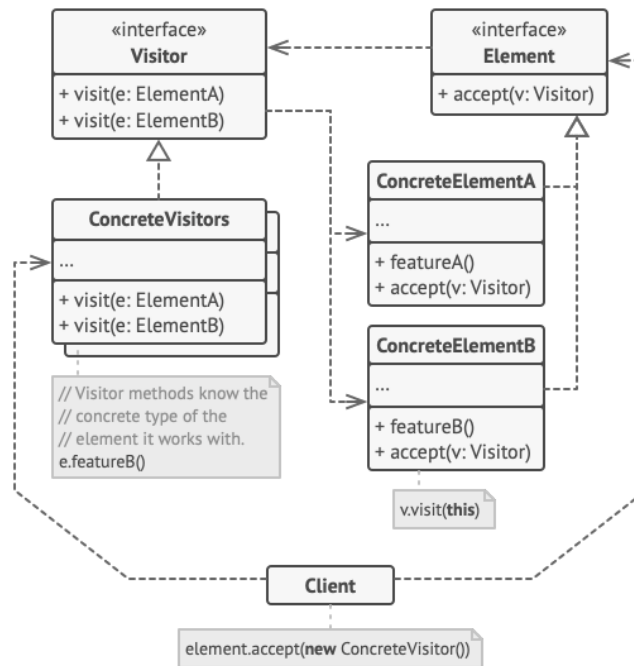
1. در نمودار کلاسی زیر، چه اصلی از SOLID نقض شده است؟ به نظرتان چرا طراح همچین اشتباهی کرده؟ چه راهکاری برای رفع این اشتباه به او پیشنهاد می‌کنید؟



2. تکه‌کد زیر مربوط به سیستم مدیریت حقوق و دست‌مزد یک اداره است. کدام یک از اصول SOLID ممکن است در آن نقض شده باشد؟ توضیح دهید.

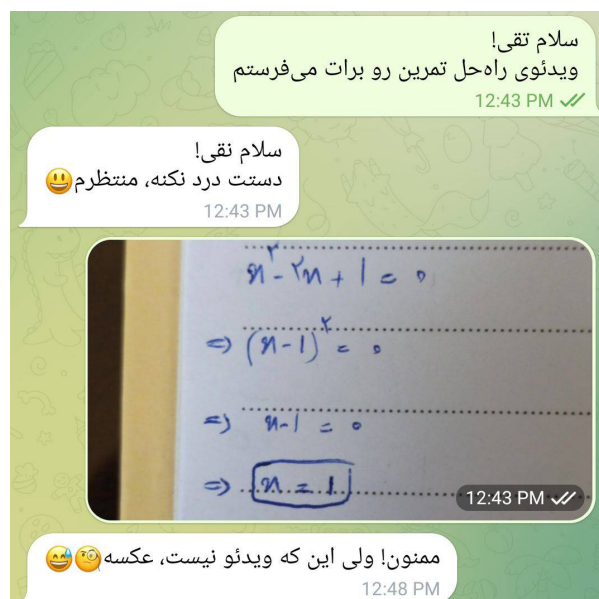
```
class Employee {
    function calculatePay() { ... }
    function reportHours() { ... }
    function save() { ... }
}
```

3. الگوی iterator یکی از الگوهای طراحی شیء‌گرا است. ساختار این الگو به صورت زیر است:



به نظرتان کدام دو اصل SOLID در این الگو رعایت نشده است؟ توضیح دهید.

4. قسمتی از چت نقی و تقی را در تصویر زیر مشاهده می‌کنید (سمت راست نقی و سمت چپ تقی است). نقی می‌خواسته از راه‌حل تمرینش برای تقی فیلم بگیرد و بفرستد، ولی کمی بعد به نظرش رسیده که عکس کافی است و نیازی به ارسال ویدئو نیست. به نظرتان پیام اول نقی باید چه‌گونه می‌بود که اگر به جای ویدئو، عکس یا صوت هم می‌فرستاد، باز هم ابهامی برای تقی پیش نمی‌آمد؟ نقی کدام اصل SOLID را رعایت نکرده است؟



سوال پنجم: معماری میکروسرویس (۵۰ + ۳۰ + ۳۰ نمره)

به عنوان یک معمار نرم‌افزار، روزی باخبر شدید که یک معماری جدید به عنوان معماری میکروسرویس (Microservice) کشف شده است. احتمالاً با ذوق و شوق به مطالعه آن پرداخته‌اید! روزی در کلاس درس مهندسی نرم‌افزار نیز اسلایدی ارائه و در آن الگوهای معماری میکروسرویس توضیح داده شد. شما که با این معماری آشنا بودید، صرفاً مفاهیم آن را مرور و دانش خود را به‌روز کردید.

در این سوال از شما می‌خواهیم که با دانش قبلی و اسلایدی که در کلاس ارائه شد، سیستمی دلخواه با استفاده از معماری میکروسرویس طراحی کنید:

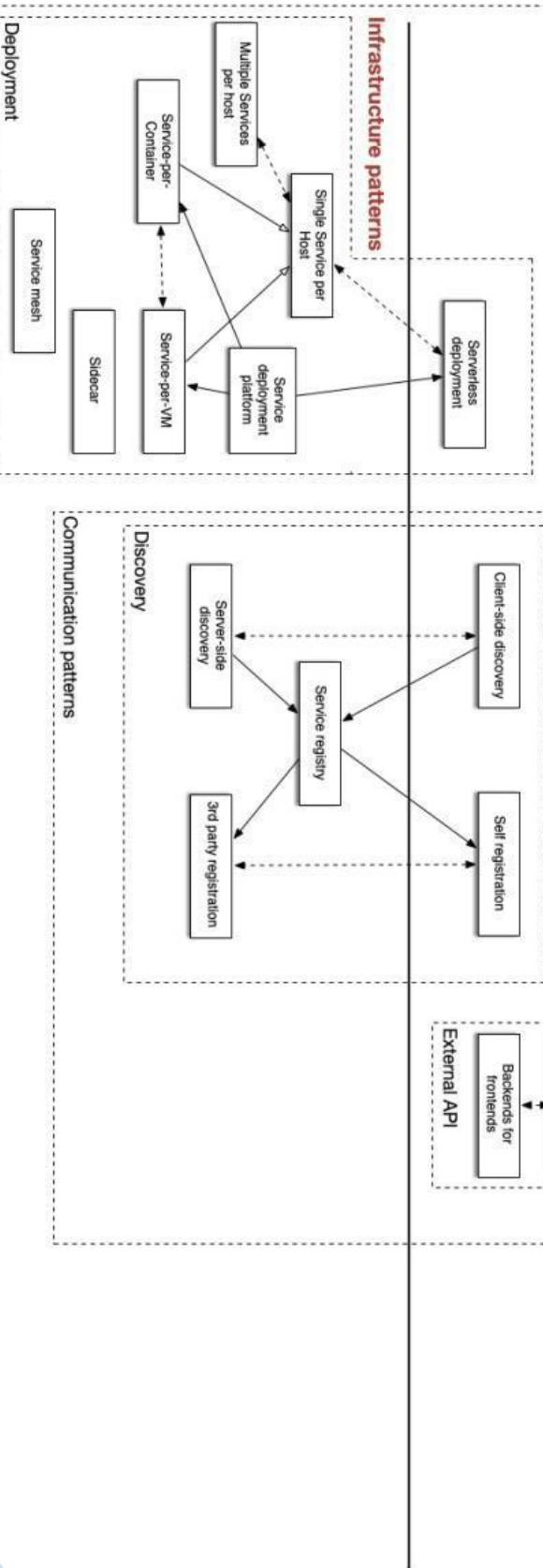
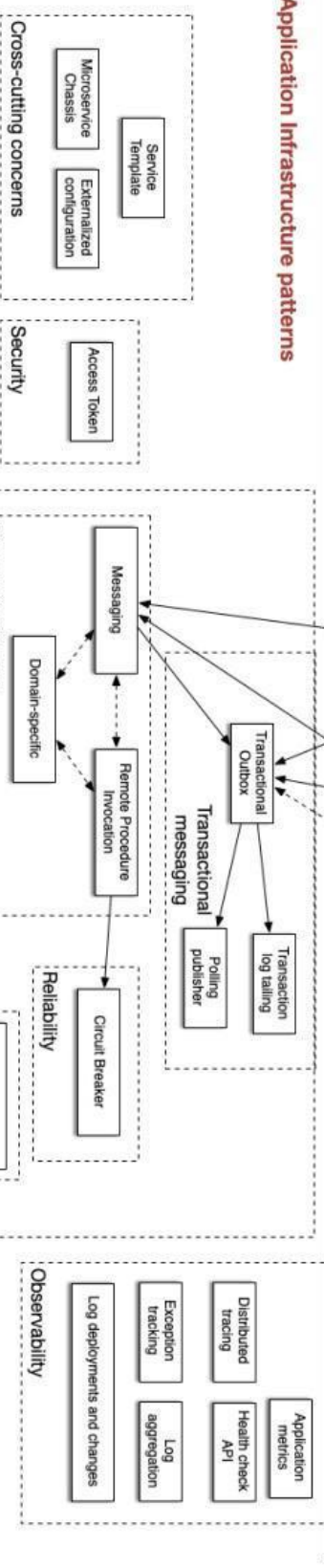
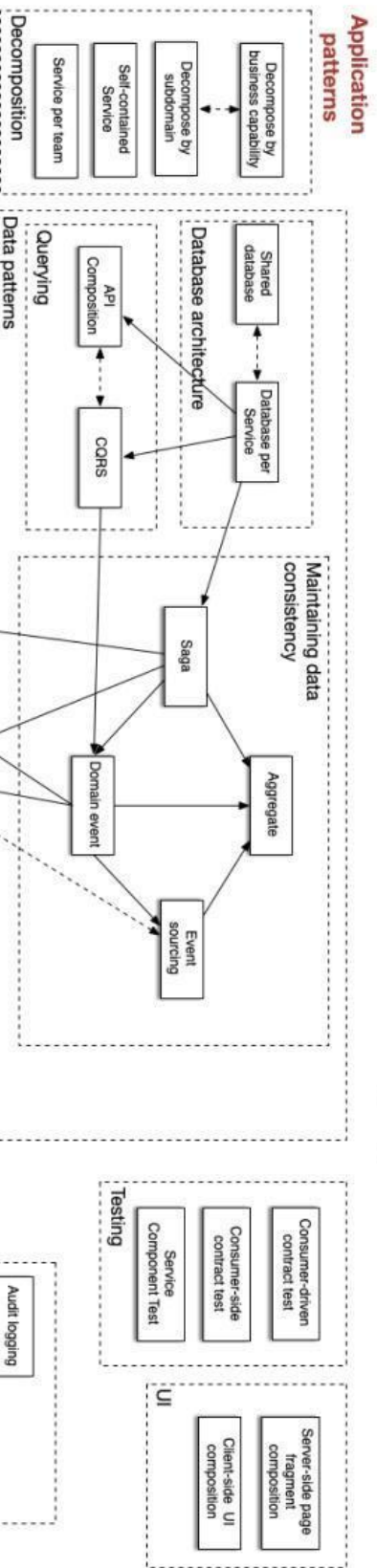
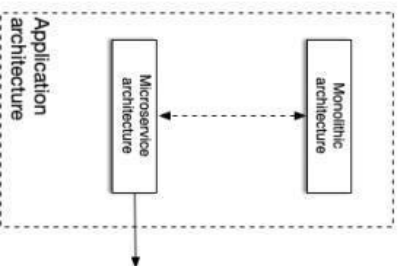
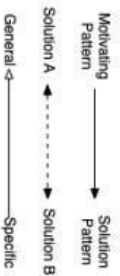
1. ابتدا از شما می‌خواهیم دست به قلم شوید و شکلی کلی از معماری سیستم دلخواه خود را رسم کنید. می‌توانید اجزا مختلف سیستم را با مستطیل ترسیم کرده و با فلش‌هایی به هم متصل کنید. لزومی ندارد که از زبان مدل‌سازی خاصی (مانند UML) استفاده کنید.

2. در طراحی معماری بالا، شما یک سری «تصمیمات معمارانه» گرفته‌اید. سه مورد از این تصمیمات معمارانه‌ی را توضیح دهید؛ مثلاً در این بخش می‌توانید بنویسید که به فلان دلیل از saga استفاده کردم.

3. در مورد سه تا از مفاهیمی که به صورت کلی در معماری میکروسرویس وجود دارد، توضیح دهید؛ مثلاً می‌توانید بگویید Service per VM به این مفهوم است که
لطفاً مواردی را که در بخش دو نوشته‌اید، در این بخش توضیح ندهید.

(تصویری از اسلاید مربوط به میکروسرویس که در کلاس ارائه شد، در صفحه‌ی بعد آمده است)

The Microservice Architecture Pattern Language



سوال ششم: مفاهیم کلی (۲۵ + ۲۵ + ۶۰ + ۴۰ نمره)

کپل، مپل و تپل سه تا از دانشجویان باحال دانشگاه صنعتی شریف هستند. آن‌ها از کودکی دوست داشتند که مهندس کامپیوتر باشند و در این زمینه فعالیت کنند. الان که به انتهای تحصیل خود نزدیک شده‌اند، تصمیم به راه‌اندازی کسب‌وکاری در این حوزه گرفته‌اند.

کپل تخصص خوبی در حوزه‌ی مهندسی نرم‌افزار و به ویژه متدولوژی اسکرام دارد. به‌علاوه، مفاهیم مربوط به نیازمندی‌ها را بسیار خوب درک می‌کند.

مپل طراح و معمار خوبی است و مفاهیم این حوزه را بسیار خوب درک می‌کند. به علاوه مدت‌ها برنامه‌نویسی کرده و با این مفاهیم آشنا است. او دوستی به نام خپل دارد که مفاهیم UI/UX را به خوبی می‌شناسد.

تپل اما، آزمون‌گر خبره‌ای است و با مفاهیم آزمون نرم‌افزار آشنایی زیادی دارد.

امروز کپل، مپل و تپل، برای ایجاد کسب‌وکار جدیدشان جلسه‌ای تشکیل داده‌اند. خپل هم به صورت آنلاین در جلسه حاضر شده (چرا که خواب ماند!). لطفاً در خصوص این جلسه به موارد زیر پاسخ دهید:

1. اگر این یک جلسه ایده‌آل بوده باشد، چگونه برگزار شده است؟ (هدف از این سوال بررسی جلسات اصولی است)
2. به نظر شما چه کسب و کاری پیشنهاد شده است؟ چرا؟ (هدف، بررسی نگاه شما در خصوص کسب‌وکارهاست)
3. دو مورد از پیشنهاداتی را که هر کدام از اعضا مطرح کرده‌اند، بنویسید. (در مجموع ۸ مورد) (هدف از این سوال، بررسی شناخت هر یک از حوزه‌هاست)
4. لطفاً به هریک از اعضا جلسه یک پیشنهاد برای بهترشدن در زمینه‌ی تخصصی‌اش بدهید. (هدف، شناخت رویکرد بهبود در ذهن شماست)



“Any fool can write code that a computer can understand. Good programmers write code that humans can understand.”

- Martin Fowler