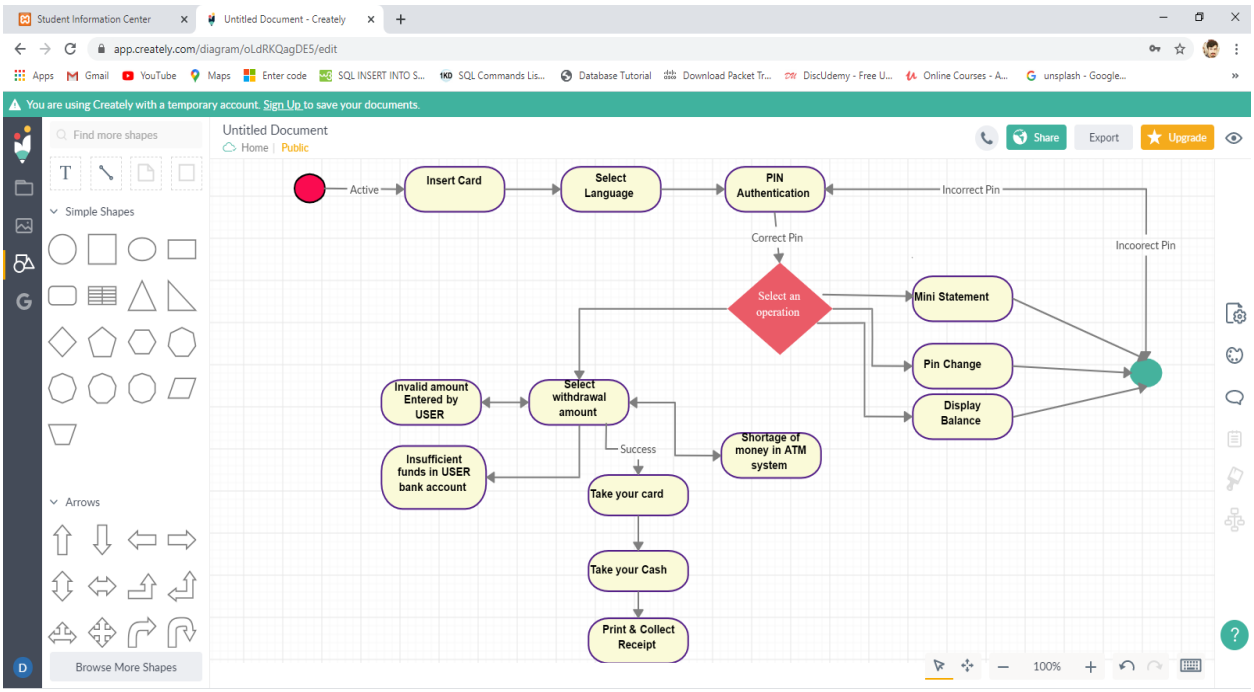
Ali Haider
14259
Final Term Exam
Course: OOSE
Instructor: Mam Sanaa Jeehan
Date: June 29, 2020

Question #1 (16 Marks)

An ATM machine can be used for Money Withdrawal and getting Mini Account statement. Make a State Machine Diagram for these two cases.

ANSWER#1:

Note:I tried to make the state machine diagram on online website.
https://app.creately.com/diagram/



Question #2 (10 Marks)

You have to make a personalized software for a consultancy firm. Some components of the software are those for which you already have code. Some parts of the software are such that you can find ready-to-be-used code from internet. If you want to embed both types of code into your product, and complete the software by coding the remaining part, what can you expect during this whole process? Will it be easy or hard? What problems you might face and how will you overcome those problems?

ANSWER#2:

Following challenges maybe occur during the whole process.

Challenge#1: An unestablished project environment is always a common challenge in terms of its impact on project delivery. If the environment is not available, then there is no way I can proceed with the project on time and under budget

Solution: To ensure efficient project development, test and pre-production environments should be made available during the development, testing, and user acceptance testing (UAT) phases.

Challenge#2: A major reason for the complexity of software projects is the constant changing of requirements. Not surprisingly, 33% of the respondents of the Stack Overflow Developer Survey consider building products with unspecific requirements, as their biggest challenge. Requirements gathering is a lot more than a handful of business consultants coming up with their ideal product – it is understanding fully what a project will deliver

Solution:

- ☞ Define and agree on the scope of the project
- ☞ Don't assume end-user needs and requirements
- ☞ Involve users from the start of existing product refurbishment
- ☞ Consider UX from the start of new product development

Challenge#3: Not reviewing code, or suppressing errors are just a means that developers use to save time and meet deadlines.

Solution: Following a formal quality assurance process is imperative for a successful launch. If you witness developers trying to cut corners in the development process, discourage it immediately. Encourage them to use best code development practices to meet the requirements sooner and more efficiently

Challenge#4: Defect identification is inevitable during functionality testing, even if the product has been through thorough unit testing during the development phase

Solution: When you come out with the test approach, scenarios, conditions, cases, and scripts, make sure your test plan covers all the requirements that are to be delivered by planning several cycles of testing

Challenge#5: product designs are under constant influence from stakeholders, the development organization, and other internal and external factors. Managing these influences is essential for maximizing the quality of systems and their related influence on future business opportunities. The increase of easily accessible, simple applications has resulted in user expectations growing exponentially.

Solution: Make sure you streamline your design and offer a consistent experience across devices, operating systems, and form factors.

Challenge#6: Integrating third-party or other custom applications, such as our ERP systems, website, or inventory management database adds substantial complexity to our project. And the bigger challenge with integration is that they remain hidden throughout the development process, and surface only at the end, leading to extra costs, delays, lowered quality, and sometimes even failure of the project.

Solution:

- ☞ Get a clear understanding of end-user requirements
- ☞ Implement an enterprise-wide framework for the platform structure of the application
- ☞ Discover and research new technologies
- ☞ Design and develop new solutions
- ☞ Test and evaluate ideas to ensure optimum integration

Challenge#7: Very often multi-tasking might give me more trouble than expected. Resources cannot focus on a single task or module if their manager bombards them with tasks.

Solution: One obvious way to be an excellent planner is to leverage project management tools like Project Pro in O365 and keep projects, resources, and teams organized and on track.

Challenge#8: Testing a software system in a controlled environment is difficult since the user is not immersed in a completely realistic working situation. It's impractical to gauge how a user will really use the application in different situations on a regular basis until it's deployed.

Solution: Testing the software, application or product in a separate real-life test environment is critical to your software's success. This will allow you to see what is working well and what is working poorly in a vacuum vs real-life use.

Challenge#9: Security breaches are on the rise; a recent study estimates that 96% of all web applications contain at least one serious vulnerability. How do I cope with evolving security threats? How do I keep each layer of your software or application secure?

Solution:

- ☞ Look beyond technology to improve the security of your software
- ☞ Develop software using high-level programming languages with built-in security features
- ☞ Require security assurance activities such as penetration testing and code review

Question #3 (24 Marks)

1. What is the difference between a Task and a Work Product? Explain in your own words and give an example of each.

ANSWER#3(1):

Task:

A task represents an atomic unit of work that can be managed: A manager assigns it to a developer, the developer carries it out, and the manager monitors the progress and completion of the task. Tasks consume resources, result in work products, and depend on work products produced by other tasks.

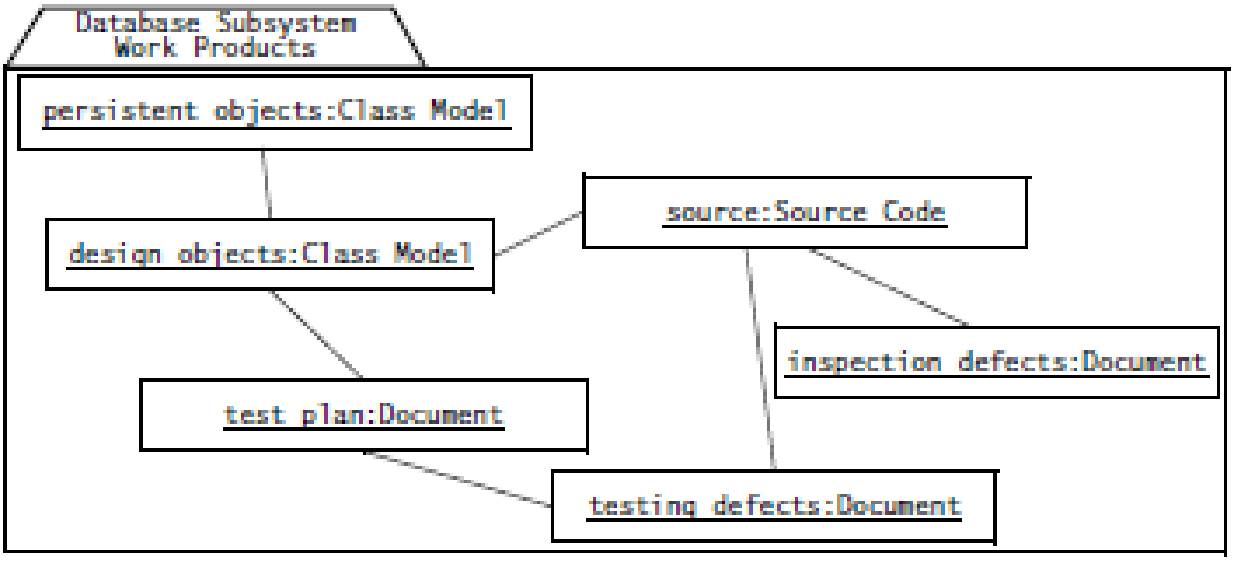
Example:

- **Develop "Out of Change" test case for TicketDistributor**
- **Review "Access Online Help" use case for usability**

Work Product:

A work product is an artifact that is produced during the development, such as a document or a piece of software for other developers or for the client. We refer to a work product for the project's internal consumption as an internal work product. We refer to a work product that must be delivered to a client as a deliverable. Deliverables are generally defined prior to the start of the project and specified by a contract binding the developers with the client

Example:



2. You are working on coding of a software in which a lot of calculations are involved. The calculations are quite easy to be done, but you do expect some inconsistency in the calculations because of some inner problem with the software. Would you rather go for Fault Avoidance technique or Fault Detection technique in the Testing Phase? Explain your answer.

ANSWER#3(2):

Fault Detection technique:

I prefer to Fault detection techniques, such as debugging and testing, are uncontrolled and controlled experiments, respectively, used during the development process to identify erroneous states and find the underlying faults before releasing the system. Fault detection techniques assist in finding faults in systems, but do not try to recover from the failures caused by them. In general, fault detection techniques are applied during development, but in some cases they are also used after the release of the system.

Fault detection techniques include testing and reviews.
Testing is a fault detection technique that tries to create failures or erroneous states in a planned way. This allows the developer to detect failures in the system before it is released to the customer. Note that this definition of testing implies that a successful test is a test that identifies faults
Review:
A review is the manual inspection of parts or all aspects of the system without actually executing the system.
There are two types of reviews: **walkthrough** and **inspection**.

In a code **walkthrough**, the developer informally presents the API (Application Programmer Interface), the code, and associated documentation of the component to the review team. The review team makes comments on the mapping of the analysis and object design to the code using use cases and scenarios from the analysis phase.
An **inspection** is similar to a walkthrough, but the presentation of the component is formal. In fact, in a code inspection, the developer is not allowed to present the artifacts (models, code, and documentation).

3. There are different types of testing. State which techniques are performed by the developer and which of them are performed by the client.

ANSWER#3(3):

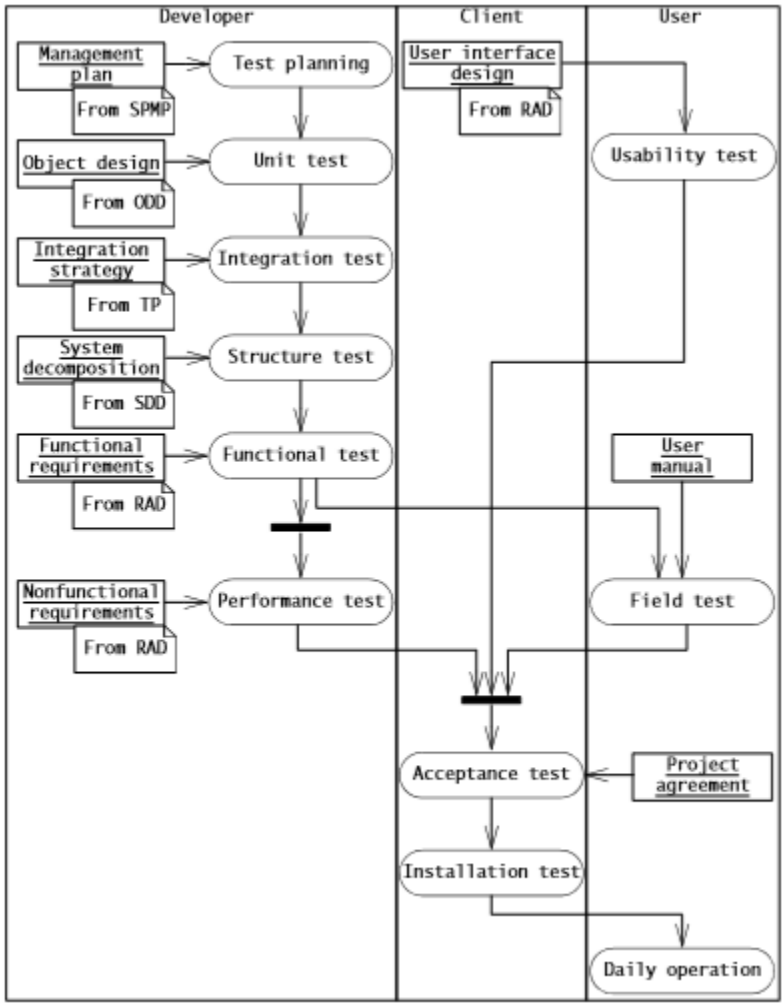
Types Of Testing:

- ☞ Test planning
- ☞ Usability testing
- ☞ Unit testing
- ☞ Integration testing
- ☞ Structural testing
- ☞ System testing

- installation testing
- Acceptance testing
- Performance testing
- Functional testing

Techniques Performed by developer:	**Techniques Performed by Client:**
☞ Test planning ☞ Unit testing ☞ Integration testing ☞ Structural testing ☞ Functional testing ☞ Performance testing	☞ Usability testing ☞ Field test ☞ Acceptance testing ☞ installation testing ☞ Daily operation

Following UML Activity diagram is showing that which technique is performed by whom.



Instructions for Paper Submission

1. Write your names and Ids at the top of each paper of answer sheet.
2. Scan / Take Photo of each paper and save each photo with a number. E.g. photo of page 1 of answer sheet be saved with name 1.jpg, then 2.jpg and so on.
3. Put all answer photos in a folder, name the folder with your Roll Number, Name and Subject Name, e.g. "11512 - Sanaa Jeehan – OOSE". Alternately, you can also make a PDF file of all the pictures and name it as explained.
4. Zip the folder and upload.