Name: Ammad Iftikhar

Reg No. # 4277-FBAS/BSSE/F21/A

Question No. 01:

Waterfall Model

The term was first introduced in a paper published in 1970 by Dr. Winston W. Royce and continues to be used in applications of industrial design. The Waterfall Model is a classical model used in System Development Life Cycle (SDLC) to create a system with a linear and sequential approach. It is also referred to as a linear-sequential life cycle model. It is termed as waterfall because the model develops systematically from one phase to another in a downward fashion.

Stages of Waterfall Model:

It has following stages…

- Requirements
- Analysis
- Design
- Coding
- Testing
- Deployment
- Maintenance

Disadvantages of Waterfall Model:

- Design is not adaptive; often when a flaw is found, the entire process needs to start over.
- Does not consider error correction.
- Reduces efficiency by not allowing processes to overlap.

Advantages of Waterfall Model:

- Reinforces good coding habits to define before design and then code.
- Allows for early design or specification changes to be made easily.
- Clearly defines milestones and deadlines.

Question No. 02:

Stages of software development lifecycle (SDLC)

1. **Planning & Requirement Analysis Phase**
   In the Planning phase, project leaders evaluate the terms of the project. This includes calculating labour and material costs, creating a timetable with target goals, and creating the project's teams and leadership structure.
2. **Requirement Definition Phase**
   Defining requirements is considered part of planning to determine what the application is supposed to do and its requirements.
3. **Designing Phase**
   Based on the requirements in SRS desired features and operation in detail are specified and documented in a DDS (Design Document Specification).
4. **Development Phase**
   In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage.
5. **Test Phase**
   This stage refers to the testing of the product where products defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.
6. **Deployment Phase**
   Once the product is tested and ready to be deployed it is released formally in the appropriate market.
7. **Maintenance Phase**
   Maintenance: What happens during the rest of software's life: changes corrections, additions and more.

Question No. 03:

User level requirement:

Proper place given on the screen to enter the book borrower's details.

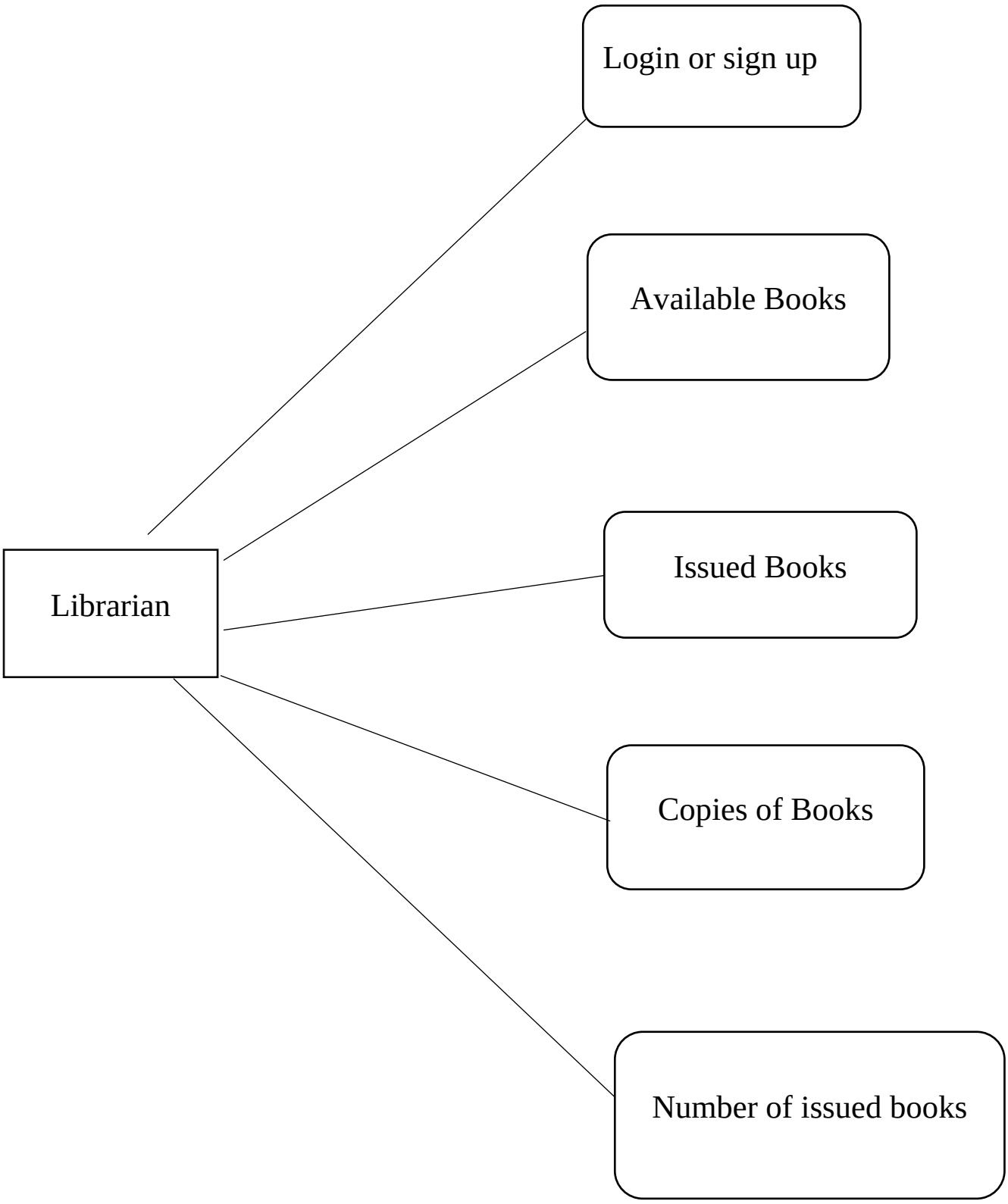
System Level Requirements:

- A book must have an ISBN.

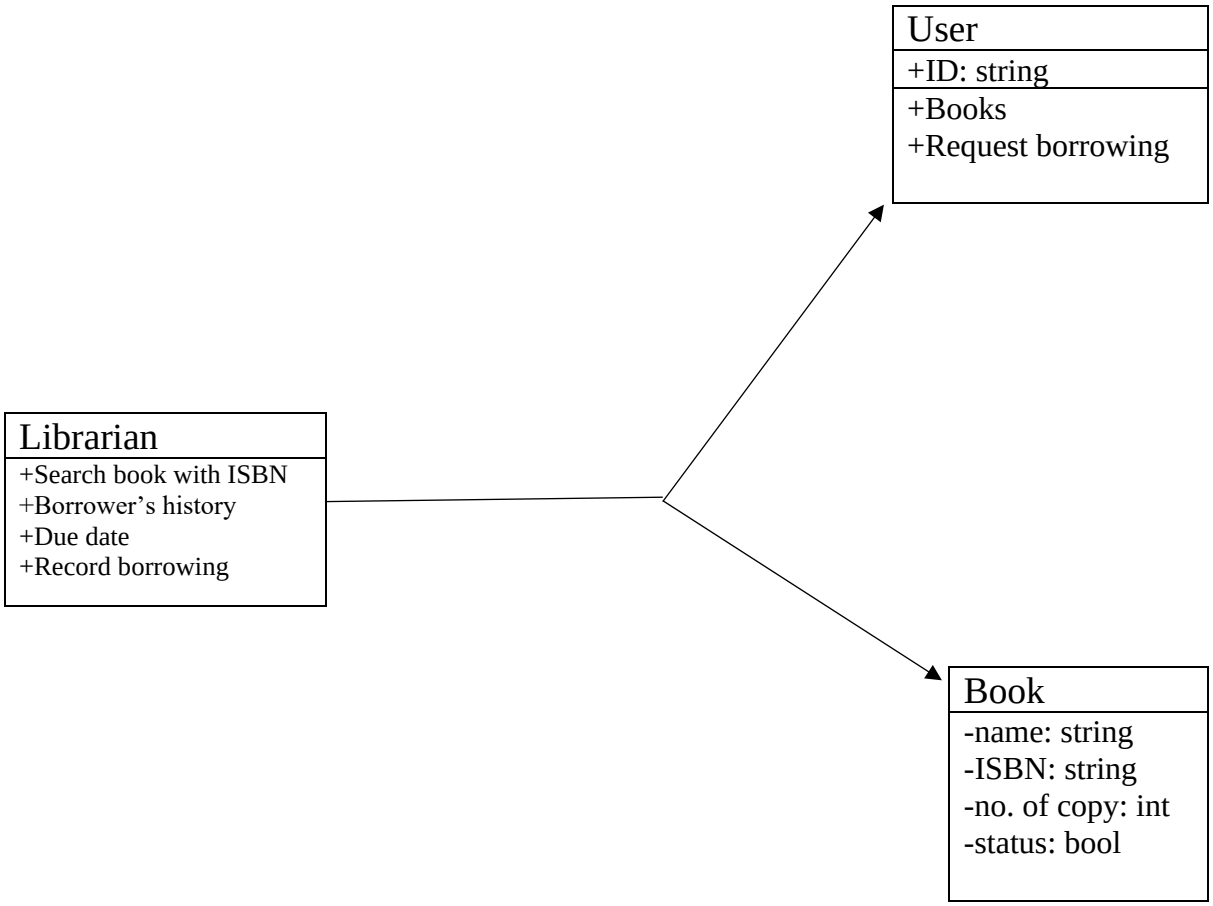
- The copy number of book must be written on it.
- Status of the book (either available or borrowed).
- Id of the borrower must be given.

Question No. 04:

Use Case Diagram:



Domain Model:



Question No. 05:

Non-functional requirements for a bike racing game:

- Ease of use.
- Secure the users achievements.
- Performance.
- Secure platform for in game transaction.

- Good frame rate for every possible low-end platform.
- Frequent updates to enhance user experience.

Non-functional requirements for online banking system:

- **Performance**

  The banking system should be up to the mark in terms of performance. The user should not face any delay in transactions or wait for any response time.

- **Usability**

  The website and apps owned by banks should be user friendly that even a person with not that much technical background could use their service.

- **Availability**

  The banking website and apps should be available 24/7 for its users for any kind of transaction if not it might lose some users.

- **Security**

  The bank should have a secure server that should contain multiple firewalls that will protect it and people money from unwanted users or any intruders.