1. **Jelaskan perbedaan metode pengembangan aplikasi model Waterfall dibandingkan dengan  
   model Agile. Jelaskan keunggulan dan kelemahannya. (UK1/TRS)**

Jika waterfall berfokus pada perencanaan dan eksekusi yang berurutan, agile lebih  
mengutamakan kelincahan dan kecepatan

Waterfall :  
• Relatif mudah dimengerti dan mudah digunakan.  
• Requirement dari sistem bersifat stabil.  
• Baik dalam manajemen kontrol.  
• Bekerja dengan baik ketika kualitas lebih diutamakan dibandingkan dengan biaya dan jadwal/deadline.  
Kekurangan dari metode ini:  
• Semua kebutuhan sistem harus diketahui di awal  
• Integrasi sekaligus di akhir sistem.  
• Testing hanya dilakukan pada setiap akhir fase, bahkan tidak jarang testing hanya dilakukan di akhir-akhir  
proyek.  
• Membutuhkan waktu yang cukup lama meski kadang proyeknya tidak terlalu besar.  
• Perubahan requirement dapat merubah keseluruhan proses yang telah dilaksanakan.  
Agile  
Kelebihan metode ini:  
• Proses Iterative dan Incremental.  
• Requirement dapat berubah sewaktu-waktu.  
• Pelacakan requirement dengan melihat Backlog produk.  
• Keterlibatan user secara aktif.  
• Rilis yang lebih cepat dan berkala, fungsi dirilis setiap akhir iterasi.  
• Testing dilakukan setiap saat.  
Kekurangan dari metode ini:  
• Interaksi dengan client yang kadang terlalu berlebihan.  
• Agile sulit diimplementasikan dalam proyek yang berskala besar.  
• Waktu perencanaan proyek yang singkat.  
• Membutuhkan manajemen tim yang terlatih

1. **Bagaimana yang harus diperhatikan interaksi program (user interaction) memiliki usabilitas yang baik**
2. *Mudah untuk digunakan*
3. *Efisien*
4. *Tingkat kesalahan*
5. *Error Recovery*
6. **Bandingkan 3 algoritma sortir untuk array antar lain buble sort, insertion sort, merge sort mana yang tercepat?**

Merge Sort

1. **Bagaimana langkah-langkah inisiasi object pada program utama**

Import kelas: Import kelas yang akan digunakan dengan menggunakan perintah require atau include.

Buat objek: Buat objek dari kelas yang telah di-import dengan menggunakan perintah new.

Panggil metode objek: Panggil metode objek yang telah dibuat dengan menggunakan notasi ->. Metode objek tersebut dapat menerima argumen.

1. **Bagaimana cara menggunakan library pada program berbasis java dan berikan contoh**

Library dalam java kita menggunakan syntax import. Syntax import digunakan  
untuk mengimport/memasukan method-method dari class/library yang lain,  
sehingga method tersebut dapat digunakan pada class yang memanggilnya.  
Contoh: import java.util.Scanner;

1. **Bagaimana langkah-langkah menampilkan data, menambah , menghapus dan  
   update dari database mysql ke dalam program atau aplikasi menggunakan JDBC?**

Import library JDBC: Impor library JDBC ke dalam proyek Anda.

Buat koneksi database: Buat objek koneksi database dengan mengisi informasi koneksi seperti nama host, nama database, username, dan password.

Buat statement: Buat objek statement untuk mengeksekusi query SQL.

Eksekusi query SQL: Eksekusi query SQL yang diinginkan, misalnya SELECT, INSERT, UPDATE, atau DELETE.

Ambil hasil query: Ambil hasil query dari objek ResultSet dan tampilkan pada program atau aplikasi.

Tambahkan data: Untuk menambahkan data, buat query INSERT dan jalankan query tersebut.

Hapus data: Untuk menghapus data, buat query DELETE dan jalankan query tersebut.

Perbarui data: Untuk memperbarui data, buat query UPDATE dan jalankan query tersebut

1. **Bagaimana melakukan dokumentasi perangkat lunak dalam implemetansi metode untuk suatu model usaha dalam bidang pendidikan?**

* Melakukan identifikasi kode program
* Membuat dokumentasi modul program
* Membuat dokumentasi fungsi, prosedur atau method program
* Men-generate dokumentasi dengan menggunakan tools

1. **Ditemukan Bug dan memperbaiki program menjadi bagian pemogram dalam menghasilkan perangkat lunak bermutu yang sesuai kebutuhan pengguna.Cara efektif yang mana untuk dapat menemukan bug dan memperbaiki program tersebut.?**

Testing: Melakukan testing secara menyeluruh dan sistematis untuk menemukan bug. Ada beberapa jenis testing seperti unit testing, integration testing, system testing, acceptance testing, dan regression testing.

Debugging: Melakukan debugging untuk menemukan dan memperbaiki bug yang ditemukan. Dalam debugging, developer dapat menggunakan tools debugging seperti IDE, debugger, logging, atau console output.

Code review: Melakukan code review oleh tim developer lain untuk menemukan bug yang tidak terdeteksi selama testing dan debugging. Code review juga dapat membantu menemukan best practice dan memperbaiki code quality.

Refactoring: Memperbaiki program dengan melakukan refactoring pada code yang rumit, tidak efisien, atau sulit dipahami. Refactoring dapat membantu meningkatkan maintainability, readability, dan scalability dari code.

Monitoring: Memantau aplikasi secara real-time untuk mengidentifikasi bug yang terjadi pada produksi. Monitoring juga dapat membantu memperbaiki performa aplikasi dan menemukan bug yang sulit terdeteksi pada tahap testing.

Continuous integration dan deployment: Mengimplementasikan CI/CD pada proses pengembangan aplikasi untuk memastikan code selalu terintegrasi dan terdeploy dengan baik. Dengan demikian, bug dapat ditemukan dan diperbaiki sejak dini sebelum aplikasi diluncurkan ke production.

Customer feedback: Mengumpulkan feedback dari pengguna untuk menemukan bug dan memperbaiki aplikasi berdasarkan kebutuhan pengguna. Dengan memperhatikan customer feedback, developer dapat memastikan aplikasi yang dibuat sesuai dengan kebutuhan dan preferensi pengguna.

1. **Jika terjadi kesalahan dalam implementasikan pengembangan program aplikasi yang terkait kebutuhan pengguna, bagaimana mengatasinya? Jelaskan tools apa saja yang dapat digunakan untuk melakukan uji coba perangkat lunak?**

Identifikasi masalah: Cari tahu apa masalahnya dan di mana terjadinya masalah. Dengan cara ini, Anda dapat memfokuskan upaya pada area yang tepat.

Buat daftar tindakan perbaikan: Buat daftar tindakan perbaikan yang harus diambil untuk memperbaiki masalah. Prioritaskan daftar ini berdasarkan tingkat kepentingan dan efektivitasnya.

Implementasikan tindakan perbaikan: Implementasikan tindakan perbaikan yang sudah ditetapkan untuk memperbaiki masalah.

Uji ulang: Setelah melakukan perbaikan, uji ulang aplikasi untuk memastikan bahwa masalah sudah diperbaiki dan tidak muncul lagi.

Tools yang dapat digunakan untuk melakukan uji coba perangkat lunak antara lain:

Unit Testing Framework: Digunakan untuk menguji fungsi atau metode individu pada program dan memastikan bahwa mereka berfungsi dengan baik.

Integration Testing Tool: Digunakan untuk menguji bagaimana beberapa bagian dari program berinteraksi dan bekerja bersama.

Performance Testing Tool: Digunakan untuk menguji kinerja program dalam kondisi tertentu dan memastikan bahwa program dapat menangani beban kerja yang tinggi.

Regression Testing Tool: Digunakan untuk memastikan bahwa perubahan yang dilakukan pada program tidak merusak fitur yang sudah ada sebelumnya.

Penetration Testing Tool: Digunakan untuk menguji keamanan aplikasi dan memastikan bahwa tidak ada celah keamanan yang dapat dimanfaatkan oleh peretas.