A building society has a long history of implementing computer-based information systems to support the work of its branches. It uses a proprietary structured systems analysis and design method. It has been decided to create a computer model of the property market. This would attempt, for example, to calculate the effect of changes of interest rates on house values. There is some concern that the usual methodology used for IS development would not be appropriate for the new project.

(a) Why might there be this concern and what alternative approaches should be considered?

(b) Outline a plan for the development of the system which illustrates the application of your preferred methodology for this project.

- a) There might be this concern because the **usual methodology used for IS** development, which is structured systems analysis and design method, is designed for developing **traditional information systems**, **not complex systems such as a** computer model of the property market. This model is likely to involve data analysis, statistical modeling, and simulations, which require a different approach. Alternative approaches that should be considered include data science and machine learning methodologies, which are better suited for handling complex data and building predictive models.

All this suggests that the 'traditional' IS development approach based on a **waterfall framework would be unsuitable. A more iterative prototyping approach is sensible**. Maybe we can think of SCRUM it can work for these kind of projects. Based on new models coming out maybe we need an agile and scrum to work with this model.

- **Define objectives: A**ctual needs of management from the model.
- **Survey existing software solutions** and the literature on the subject. Has anyone done this already? Or the model is already in the market.
- **Consult experts.** JAD (joint application development) sessions are mentioned in text as one possible approach. There are many other techniques to elicit relevant knowledge from domain experts which should be considered.
- Based on the above draw up an **initial prototype design.**
- **Build prototype.**
- **Test it**, probably using real data from the recent past; you can compare the results of the model with what actually happened.
- Analyze the causes of discrepancies; modify the model and rerun.
- Stop this cycle when you run out of time, or are not getting significant improvements.
- Carry out a general validation of how accurate the model is, and how easy it is to use

Question 2:

A software package is to be designed and built to assist in software cost estimation. It will input certain parameters and produce initial cost estimates to be used at bidding time.

(a) It has been suggested that a software prototype would be of value in these circumstances. Explain why this might be.

(b) Discuss how such prototyping could be controlled to ensure that it is conducted in an orderly and effective way and within a specified time span.

Answer

a)

Prototyping helps in creating a functional design of the product to be created before it is actually created. Hence, this prototype can be used by investors, testers, demonstrators, developers, etc. to visualize the end product. It helps in understanding the features that may need modifications, this helps in perfecting the end product.

The reason for doing it in this scenario is that it will promote real innovation with less cost and less risk of failure. It will allow quickly weeding out the ways that are not working. It will also help in understanding user requirements and gain an understanding of the problem in a technical sense.

b) The most important benefit of working with prototypes is that they help in simulating real-world scenarios. **They give an idea of the future product. It should communicate the needs, demands, and ideas of how the final product will work and behave**. The process can be **controlled by reviewing the prototype and revising any changes needed.** It helps in enhancing the prototype. The central testing issue should be resolved. This will help in not losing focus or not getting bound to certain trial and error situations. Review from the user help in knowing how to revise the prototype in a better way.

Question 3:

An invoicing system is to have the following transactions: amend invoices, produce invoice, produce monthly statements, record cash payments, clear paid invoices from database, create customer records and delete customers.

1. What physical dependencies govern the order in which these transactions are Implemented?

2. How could the system be broken down into increments which would be of some value to the users (hint - think about the problems of taking existing details onto a database when a system is first implemented)?

Answer

1. The physical dependencies are as such that the **invoices are arranged and managed** in an organized fashion and are updated as soon as the transaction or any operation takes place.

- The recorded cash payment will only be deleted once the payment is made
- the customer fields cannot be eliminated before it is created
- the monthly statement cannot be created before the payment is recorded.

2.

The implementation should be in the following format:

1. Create Customer rows/ records
2. Initiate invoice
3. Maintain the records of cash payment
4. Amend invoice time to time
5. Produce monthly invoice
6. Clear paid invoice from the database
7. Delete customer

Questions 5:
In a college environment, an intranet for students that holds information about courses, such as lecture programmes, reading lists and assignment briefs, is often set up. As a 'real'exercise, plan, organize and carry out a JAD session to design (or improve the design of) an intranet faciltY.
This will require:

- preliminary interviews with representative key stakeholders (for example, staff who might be supplying information for the intranet);
- creation of documents for use in the JAD proceedings;
- recording of the JAD proceedings;
- creating a report which will present the findings of the JAD session.