PART-1
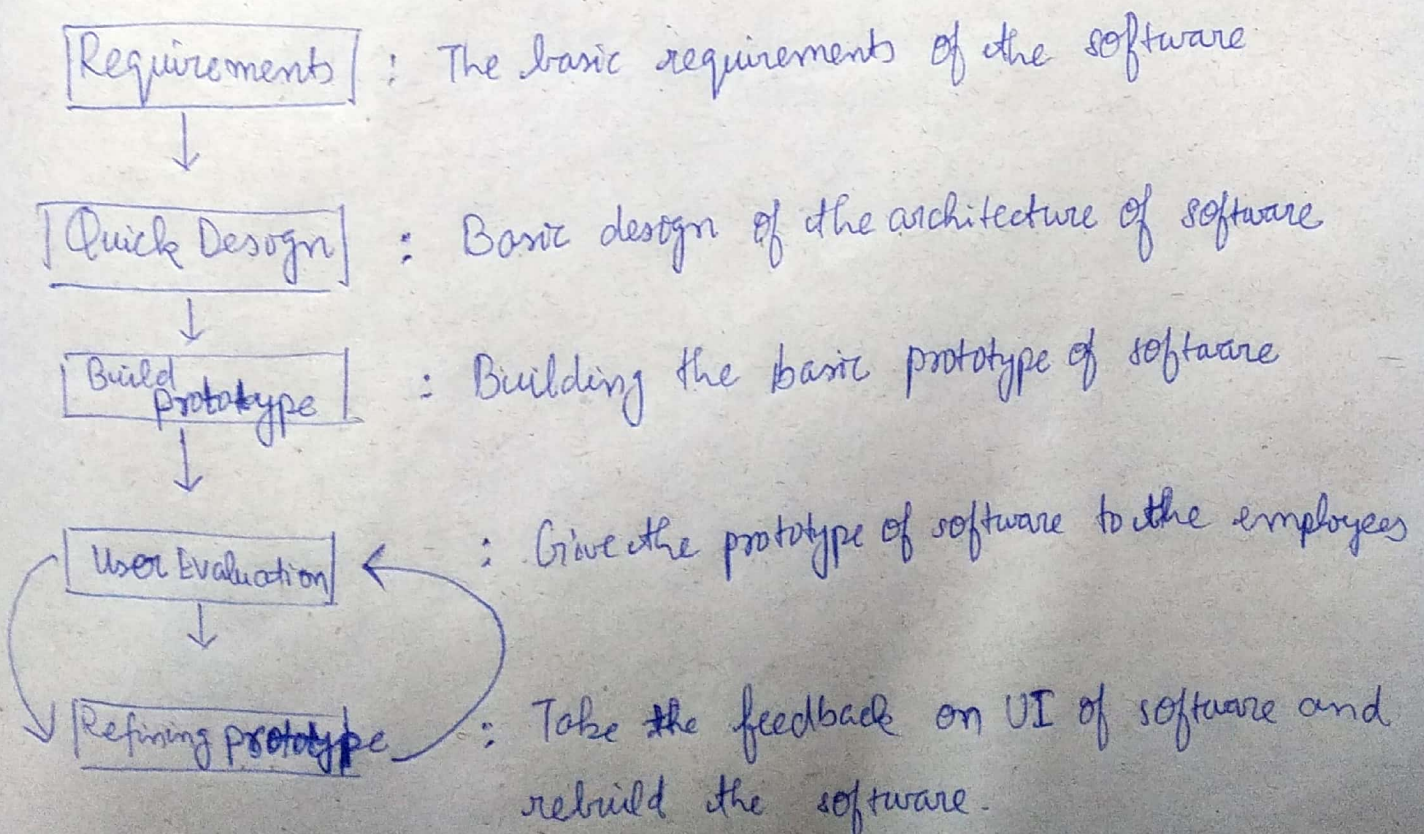
Q1. I think spiral model of development ~~can~~ should not be used to develop that bookkeeping software. It is because;

→ The project is an internal project and is not very large.

→ Frequent releases are not released

→ As it is an internal project, risks of cost evaluation isn't much important.

→ This software isn't a client software. Thus it isn't a high-risk project.

In this case, I think prototyping model will be the best model.

The UI is currently unsure as they ~~it~~ are unaware of the requirements of the stakeholders properly.

| Requirements | : The basic requirements of the software

↓

| Quick Design | : Basic design of the architecture of software

↓

| Build Prototype | : Building the basic prototype of software

↓

| User Evaluation | ← : Give the prototype of software to the employees

↓

| Refining Prototype | : Take the feedback on UI of software and rebuild the software.

Q2. In this case, waterfall model should be used to develop the payroll application.

It is because:

→ clearly defined stages of production as it is already developed multiple times

→ Well understood milestones for the team

→ Easy to arrange tasks as requirement modules are clearly known.

→ Process & results are already well documented.

→ Easy to manage the project. Hence, resources are utilized for that project.

Case 2: Online Auction System (PART-2)

Functional Requirements:

→ **New User Registration:** Any user who is new to the system can register himself/herself to the system by abiding to the terms & conditions specified by the system.

→ **User login/Authentication:** If a user tries to put items in the system for auction or if the user tries to place a bid for an item, the system checks if the user is authenticated/logged in state, and if he/she isn't, he is taken to the login page to authenticate themselves.

→ ~~Search~~ **Adding Items into System:** User after authentication can add new items into the portal for selling.

→ **Placing Bid:** An authenticated user can place a bid after selecting the items he/she wants to buy.

→ **Payment Service:** 3rd party payment APIs help in and abstract most of the payment details. Just entering the credit card/debit card details after confirmation from the seller deducts the money safely and generates a receipt.

→ Shipping & Transportation: The seller looks after the delivery of the product to the customer. The seller tracks at each step of delivery and ensures that delivery is flaw-less and shipped to the door step.

Non-Functional Requirements

a) Performance Requirements:
  → System should be highly available. (24×7).
  → At least 10K users should be able to place bid at a particular time. Should be highly scalable infrastructure.

b) Interface Requirements:
  → Site must be accessible from any standard browser i.e. Chrome, Opera, Mozilla firefox
  → HTML5 should be used for developing that-site

c) Security Requirements
  → The user's password must be salted multiple times and stored in secured database.
  → Only authenticated users should be able to place bid.

d) Database Requirements:
  → Available database (24×7)
  → Horizontally Scaled Database with all ACID properties present.