1. Software Development Life Cycle

2. System Development Life Cycle

3. Analysis, Design, Implementation, Testing, Deployment, Maintenance

4. Analysis – Thinking of an idea for the project as well as feasibility testing to test if the idea itself can work.

Design – Preparing the idea and turning it into a game design document, story boarding if necessary as well as list the design of the games to prepare for the implementation.

Implementation – The phase that is mainly about coding and creating the functions that will turn the idea into a product.

Testing – if it’s a game then getting beta testers to test out the game before the game is officially released to find bugs that could have possibly been found that wasn’t found before as well as to see if the product fits with the original needs of the customer/users.

Deployment – The actual selling of the product to regular customers and/or potential new customers.

Maintenance - When the actual bugs that couldn’t be found in the testing itself is when there’ll be periodic maintenance if necessary to fix bugs that happened to the users from the products.

5.

Analysis – Requirement Specification Document

Design – system design specifications

Implementation – Source Code

Testing – Test plan

Deployment – The sales count

Maintenance – number of problems that popped up during the time of the products usage

6. Program integration is the combination of multiple programs or parts to function together or work a one program

7. Big Bang Model, Waterfall Model

8. Waterfall model steps are easy to follow and understand, provides structure to the less inexperienced personnel and good for management control

9. All the requirements must be known beforehand, You can’t move on to the next step until the previous one is done, can give a false impression of progress

10. Simple Model, little to no planning required and easy to manage

11. High risk of uncertainty, not a good model for more complex object oriented projects and a poor model for long and ongoing projects

12.It is the less risky approach wherein the components and subsystems are integrated as they are being developed into several working mini-builds of the system.

12. The product can be split into several builds as the product itself is being made that can be integrated individually and can show the potential customers a prototype before the final product has been made