|  |  |  |  |
| --- | --- | --- | --- |
|  | **Waterfall** | **V-model** | **Agile** |
| *Features* | | | |
| **Project Scope** | Working only for small scope, step by step (linear-sequential) | Medium, parallel testing | High, all stages are performed in parallel |
| **Change Tolerance** | Absent | Low | High |
| **Planning Scale** | 10/10 | 8/10 | 3/10 |
| **Team Size** | Minimum (which team works in another stages) | Medium (team works in another stages + QA engineers) | High (Each team works simultaneously and in parallel) |
| **Customer Involvement** | Minimal or absent | Minimal or absent | High |
| **Feature Prioritization** | Present, all consistent with the documentation | Present, all consistent with the documentation | Step by step, during project creation and development |
| **Risk Management** | Risk of a change | Ability to prevent risk at every stage of software development | The risks are minimal, the changes are welcomed |
| **Documentation Importance** | High | Middle | Minimal |
| **Communication & Team Involvement** | Minimal | Present | High |
| *Pros & Cons* | | | |
| **Pros** | Structured, clear, well documented | Can detect defects in the process | High team interaction,you can quickly change and improve product, flexibility |
| **Cons** | Finished product in the last phase, inability to see the gradual growth of the product. Impossible to change. Not suitable for long and complex projects. | Not flexible, not suitable for long and complex projects. | Minimum documentation, dependence on experience, high dependence on customer interaction |

Please fill in the table below comparing the 3 major models of SDLC.