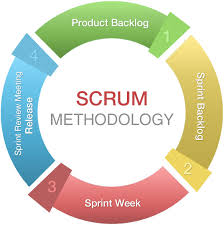
**Agile Methodology (Scrum)**



Methodologies are methods which are used to carry out projects and manage development teams in the IT industry. Many different mythologies fall under the agile mythology category. The main difference between agile methodologies and more traditional mythologies (Waterfall for example). Unlike waterfall model, agile mythologies follow a sequential approach which gives us room to change the requirements of the projects if we were to decide to aim for higher grades during the project.

This also gives us the chance to work on our project and start the very basic features without spending too much of our given time doing extensive planning, instead we can start building on the early features of the project.

This also allows us to start testing the project as we work on it, as opposed to only leaving the testing as one of the last phases, so we can make adjustments on the early features before we start working on the more advanced features as opposed to testing the whole project at once, which may lead to us fixing a large amount of errors during the final weeks of the deadline.

One of the disadvantages of agile methodology is that if it not followed consistently, the project can become stages of code sprints, which could lead us to having an incomplete final project; but the reason we are choosing an agile methodology over waterfall is, if we do decide to go for the higher grades; we can adjust the requirements and we can add in new features once we have the basic features complete.

More specifically, we are following the Scrum agile mythology. This method is more beneficial to us than other methods is because of the constant meetings we have, so we can stay up to date on what has been complete, and what still needs to be worked on. This ensures we stay on top of the project work as opposed to delaying the work and completing other modules, because we won’t have to present our work at weekly meetings if we were using waterfall methodology for example.

**Agile Lifecycle Description (Scrum)**

*Planning/Analysis Phase:*

In the very first phase of the lifecycle, our team is going to be planning and analysing extensively so we can start to work on the basics of the program as soon as the very first sprint. In this phase we plan, what our software will do, what features it should or shouldn’t have.

*Sprint Planning/Meeting:*

In this stage of the lifecycle, we assign work to the members of the team; determining what work which will be completed by each member. This will determine what work they will cover in the sprint week and present during the next meeting. Every meeting apart from the very first, we will discuss what we have completed since the last meeting, the problems we encountered and if we fixed them. Work is then assigned for the sprint.

*Sprint:*

During this phase, which is the key phase of the lifecycle, during this week; work that was assigned to the members is completed. In this phase we plan on actually creating our software, creating the actual functionalities, working on the documentation and testing the program. Each sprint will be one week, we will then return to the last phase of our lifecycle, iterating the meeting and sprint phase until the project is complete.

Working system release:

This is the final phase of our project lifecycle, this will be the point where we will upload the piece of work we have created (Software & Documentation). We will demo the product to our lab tutor. During this phase we expect to receive feedback on our project.

**Airport Booking System Moscow Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Must Haves** | **Should Haves** | **Could Haves** | **Won’t Haves** |
| Login System | Book specific Seats | Landing areas available for planes | Change plane route |
| Book Flights | View plane information (Engine size etc.) | Staff information on plane. | Delete/edit plane information (Engine Size) |
| Edit Bookings | Automatic updating flight departure board | Can a specific type of user edit bookings? |  |
| View Bookings | Arrival times of planes | How you wold like to visualise it? |  |
| Edit flight information (Destination, departure times etc.) | Who will be able to edit this information? | Would it be possible to add/remove flight fields? |  |
| Different user permissions. | What are the types of users? | Would it be possible to define new types of users? |  |
| UI | Test mode visualisation | Graphical visualisation | 3D visualisation |
|  |  |  |  |