**Blog 3**

Finite State Machine is a mathematical model used to describe and control the execution flow. The abstract machine can active be in one of limited amount of states, that change in response to the external input at any given time. According to N.Philips, FSM can be viewed as a function mapping a sequence of events into a corresponding set of output events. Regarding the fact, that the behaviour of the game character may be represented as a series of diverse “mental states” with the change driven by the action, FSM seems to be a natural choice to represent Artificial Intelligence of the game. Main advantages of this model include its speed, related with a direct access to a single array, as well as expressiveness and most importantly the fact, that it may be compiled into a compact data structure (with current state assigned to dynamic memory). Implementation of a pointer, that would direct the firmware to set of instructions in the microcontroller’s memory is an efficient, and at the same time the most professional, alternative to performing multiple *if-else* statements.

On the other side, FSM may be hard to implement if some of the activities include tricky adjectives, such as *“wait for 2 nd time xyz sees the bomb”* or *“find the closest rock”.* Hence, creatinga concise FSM model requires bit more experience in the field that it may seem to.

Last but not least, the game developer may want to consider an implementation of state diagrams instead of FSM model, as they represent the state changes in much more standardized way, e.g. by adding separators to sub states and extra operators, that creates a clear image of the system.