***Software Project Management***

Final Year Project Name: *UpSkiller*

***Abstract***

Software development organizations have a well-defined process for the design and development of the outstanding software applications, so that the human error is reduced. An absolutely dynamic team of developers is required to make project a success for the stake holders. But the progress and performance analysis of a software developer has always been done either manually or it is an ignored lemma. The proposed solution is an automated system Upskiller which is also a product of development. This system can be used to improve quality coding, time management, and human resource management; by using code review analysis, and rating analytics. It is a margarine of Crucible’s data which is processed and cleaned in Node.js before it enters Elastic search. The data is understood by the system using Hash Tagging and meaningful information is displayed through interactive interfaces over React.js. Upskiller will provide a glass door through which the development stake holders can observe the performance and progress of developers.

***Macro Processes***

1. Data Generation Module
2. Generation
3. Serving
4. Data Collection Module
5. Requesting
6. Storage
7. Data Analysis Module
8. Metric Scoring
9. Accumulative Scoring
10. Data Visualization Module
    1. Data Representation
    2. Team Builder
    3. Admin Module

***Micro Processes***

1. Data Generation Module
   1. GET APIs
   2. POST APIs
   3. Database Design
   4. Data Accessing Functions
   5. Define Entry Points
2. Generation
   1. POST APIs
   2. Populate Database
   3. Hit Entry Points
3. Serving
   1. REST APIs for requesting data.
4. Data Collection Module
   1. Request the data APIs
   2. Saving the requested data
5. Requesting
   1. Axios Requesting APIs
6. Storage
   1. Temporary Data Storage
   2. Asynchronous Operations
7. Data Analysis Module
   1. Calculate User Scores i.e. perform analysis
8. Metric Scoring
   1. Applying metric calculation on User Scores
   2. Code quality scoring
9. Accumulative Scoring
   1. Technical Accumulation of All Scorings calculated before
10. Data Visualization Module
    1. Functions displaying the information maintained till before.
    2. Data Representation
       1. React UIs for displaying data
    3. Team Builder
       1. Virtual team formation UIs for dynamics
    4. Admin Module
       1. APIs for managing Users
       2. Front Ends for the APIs designed

**Note**: *Micro Processes can be further sub divided into its constituent micro processes.*