|  |
| --- |
| agile phases: Start Date => (27-4-2023) |
| - CONCEPTION AND REQUIERMENTS  - DESIGN  . UML Design  -Data Structure Design  Error Handling Design  Performance Design  Security Design |

|  |
| --- |
| Function Requirments |
| 1. comments page to manage it. --------------------------------------------------------------- ✔  * get one comment with id * get all comments with pagination * get the latest comments and the count them * get all comments within interval of time * get all comments on that blog ordered by * get all the comments for that user * add comment * remove comment * edit comment * get the count of comments of blog  1. Reply page ------------------------------------------------------------------------------------------ ✔  * Get one reply with id * Get all replies on that comment * Add * Update * Delete * Exceptions  1. Blogs page ------------------------------------------------------------------------------------------ ✔  * one blog with id * get the blogs related to one writer * get all blogs related to that category * get the invisible blogs for that user who disappear them * display the writer blogs that need confirm * scheduling The Posts //event * get all blogs with pagination * get the most popular blogs with comments and likes // how to paginate the result? * get the blogs in interval of time * add blog * update blog * soft delete blog => when delet the blog the   media deleted too  blogtags  comments  reply  read by  reports  likes  share   * search for blog in title, body, description and with time created in it * delete blog after 10 reports from users //event * undelete the blog * invisible the blog * Visible the blog  1. media page  * display all media for that blog. * if I need to add vedio from youtupe < optional > * add * upload images * update * delete  1. tags page   • display all tags   * display one tag with id * add * update  1. Blogtags page  * display all blogs in that tag with pagination * display all Tags in that Blog with pagination * Add * Update * Delete (force)  1. User page  * display the users * search for user with name * the event is the blocking and have several listeners: 1) notify the writer 2) if it the tenth Be User * Scheduling // block for 1 month if it 10, and delet the blocks to recount again * Ask for be writter * Sign in * Log out * Register new user * Update user data * Delete user softly * Force Delete * Authentication  1. Admins page  * Get all * Get with name * Get with id * Get current admin => needs remember tokens tables like user * Login * Logout => needs remember tokens tables like user * Register new admin => email\_verified\_at column * Upload admin photo after registration * Disactivate admin // cannot log in * Activates admin * Update admin  1. Category page  * Display all category * Get one category * Search for category with name * Add category * updates * delete category softly * cascading the deletion * restore category * Force Delete category  1. Blocks page  * Get number of blocks * Add block * Unblock  1. Follow page  * Get all for that user * Add * Unfollow  1. Reade by  * Get all the user who read that blog * Add  1. Reports page  * count the # of reports on that blog * add * delete  1. Likes page  * Get all on that blog * Add * Unlike  1. Shares page  * Get all on that blog * Add * delete   BUGS:   * + - 1. Cascading on delet, restore and force delete Blog and its Relations       2. Cascading on delet, restore and force delete Category and its Relations       3. Cascading on delet, restore and force delete User and its Relations   systems need:   1. notification 🔔 2. Comment 3. Replay on his comment 4. liked 5. shared 6. followed him 7. reported 8. Blocked 9. Notify the user when his followed writer adds new blog 10. Emails  * Warn the writer that someone blocked him * Tell the writter that he cannot write any blogs for month becouse of 10 blocks he had |
| In the next version   1. Contributed Blogs => some blogs will be able to modified by some users in another branch as open source blog. 2. Suggest blogs as the interests of that user 3. Multi languages 4. SEO Matrices 5. get the reported comments =====🡺 need to add the feature of reporting on comments 6. Last log in time ⌚ 7. Who is active (log in) now? 8. Delete admin // who is gona do this? => the SUPER admin 9. ADD News page 10. Get all 11. Get one 12. Search 13. Add, show, update and delete news 14. appear or disappear news 15. scheduling The news 16. Chates system 17. Search for admin 18. Caching 19. restore the user 20. add block schema, check if the user is writer it needs to create custom ressolver for AddBlock. 21. block user from commenting // method that contain some words if it found in his comment 22. add share check if he is writer 23. add follow check if he is writer 24. add report check if the blog is hidden 25. algorithm for check on the images or vedios if it hase any thing wrong 26. page for the deleted users with who deleted him from admins 27. notify admin If user add a comment or blog or reply with blocked word 28. the business logic of ask for be a writer |

* + - 1. validate the ressolver + Exceptions
      2. Check the authentications and authorizations