**#1**

Funkcionālas un nefunkcionālās prasības, izmantojot SWEBOK sadale 1.3:

* Sistēmai jāspēj reģistrēt jaunus lietotājus.
* Sistēmai jānodrošina lietotāju autentifikācija un autorizācija.
* Lietotājiem jāvar izveidot un rediģēt savus profila datus.
* Sistēmai jāspēj apstrādāt un pārvaldīt lietotāju ievadītus dati.
* Sistēmai jānodrošina iespēja izveidot, rediģēt un dzēst satura vienības (piemēram, raksti, attēli, vai faili).
* Jābūt iespējai saņemt un pārvaldīt lietotāju atsauksmes un komentārus.
* Sistēmai jāspēj nosūtīt e-pasta paziņojumus par svarīgiem notikumiem lietotājiem.
* Jābūt iespējai meklēt saturu sistēmā, izmantojot dažādus kritērijus.
* Sistēmai jāspēj veikt sistēmas veiktspējas uzlabojumus un optimizācijas.
* Jābūt iespējai izveidot un pārvaldīt sasaistes ar citām sistēmām vai platformām.
* Drošība: Sistēma jānodrošina ar drošu autentifikācijas un autorizācijas sistēmu, lai aizsargātu lietotāju datus pret neautorizētu piekļuvi.
* Veiktspēja: Sistēmai jābūt spējīgai apstrādāt lielu apjomu datu un lietotāju pieprasījumu bez ievērojama veiktspējas samazinājuma.
* Pieejamība: Sistēmai jābūt pieejamai visu laiku vai vismaz lielāko daļu no tā, nodrošinot minimālu izpildes pārtraukumu.
* Lietojamība: Lietotājiem jābūt spējīgiem viegli izmantot sistēmu, ar intuitīvu un lietotājam draudzīgu interfeisu.
* Skalējamība: Sistēmai jābūt spējīgai pielāgoties un skalēties atbilstoši pieaugošajam lietotāju skaitam un datu apjomam.
* Uzticamība: Sistēma jāprojektē tā, lai novērstu datu zudumus vai bojājumus, nodrošinot atgūšanas mehānismus un rezerves datu kopijas.
* Atbalsts: Nepieciešams nodrošināt atbalsta mehānismus, lai palīdzētu lietotājiem atrisināt problēmas vai izteikt priekšlikumus sistēmas uzlabošanai.
* Atbilstība standartiem un regulējumiem: Sistēmai jāatbilst attiecīgajiem nozares standartiem un likumdošanas prasībām, piemēram, datu aizsardzības noteikumiem.
* Platformu neatkarība: Sistēma jābūt neatkarīgai no konkrētās platformas vai operētājsistēmas, lai varētu darboties dažādos vides.
* Dokumentācija un izstrādes pārredzamība: Nepieciešams nodrošināt detalizētu dokumentāciju par sistēmas darbību un izstrādes procesu, lai atvieglotu jaunu darbinieku ieviešanu un nākotnes izmaiņu veikšanu sistēmā.

Funkcionālas prasības, izmantojot User Stories:

* Kā lietotājs, es vēlos reģistrēties sistēmā, lai piekļūtu papildu funkcijām.
* Kā lietotājs, es vēlos pieteikties sistēmā, izmantojot savu e-pasta adresi un paroli.
* Kā lietotājs, es vēlos rediģēt savu profilu, lai atjauninātu savus kontaktdatus.
* Kā lietotājs, es vēlos izveidot jaunu rakstu, lai publicētu svarīgu informāciju.
* Kā lietotājs, es vēlos komentēt citas lietotāju rakstus, lai dalītos ar savu viedokli.
* Kā lietotājs, es vēlos saņemt e-pasta paziņojumus, kad kāds atbild uz manu komentāru.
* Kā lietotājs, es vēlos izmantot meklēšanas funkciju, lai atrastu interesējošo saturu.
* Kā administrators, es vēlos pārvaldīt lietotāju atsauksmes, lai uzturētu sistēmas kvalitāti.
* Kā sistēmas administrators, es vēlos veikt sistēmas veiktspējas testēšanu un uzlabojumus, lai nodrošinātu efektīvu darbību.
* Kā integrācijas speciālists, es vēlos izveidot API integrāciju ar trešās puses sistēmu, lai nodrošinātu datu sinhronizāciju.

Temats var būt, piemēram, "Socialā tīkla platforma hobijsportistiem". Šāds temats ļautu izstrādāt sistēmu, kas ļauj lietotājiem dalīties ar savām sporta aktivitātēm, iegūt padomus un atbalstu no citiem hobijsportistiem, izveidot notikumu plānus utt.

**------------------------------------------------------------------------------------------------------------------------------------------**

**#2**

Nosaukums: Pieteikšanās lietotnē

Aktieri: Lietotājs

Soļi:

* Lietotājs atver lietotni uz savas mobilās ierīces vai datora.
* Lietotājam tiek parādīta pieteikšanās lapa ar laukiem, lai ievadītu e-pasta adresi un paroli.
* Lietotājs ievada savu e-pasta adresi un paroli.
* Lietotājs nospiež "Pieteikties" pogu.
* Sistēma pārbauda ievadītos datus un paziņo par pieteikšanās veiksmīgumu, ja dati ir derīgi.

Alternatīvi soļi:

* Ja ievadītā e-pasta adrese vai parole nav derīga, sistēma parāda kļūdas paziņojumu un prasa lietotājam ievadīt pareizos datus.

Nosaukums: Publicēt rakstu

Aktieri: Lietotājs

Soļi:

* Lietotājs pieteicas savā kontā un noklikšķina uz pogas "Izveidot jaunu rakstu".
* Lietotājs aizpilda raksta virsrakstu un saturu tekstā.
* Lietotājs nospiež "Publicēt" pogu.
* Sistēma saglabā jauno rakstu un parāda apstiprinājuma paziņojumu.

Alternatīvi soļi:

* Ja lietotājs mēģina publicēt rakstu bez virsraksta vai saturu ievadīšanas, sistēma parāda kļūdas paziņojumu un bloķē publicēšanas procesu.

Nosaukums: Meklēt citus lietotājus

Aktieri: Lietotājs

Soļi:

* Lietotājs pieteicas savā kontā un izvēlas meklēšanas funkciju.
* Lietotājs ievada meklēšanas kritērijus, piemēram, vārdu vai intereses.
* Sistēma izpilda meklēšanu un parāda rezultātus saraksta veidā.
* Lietotājs izvēlas konkrētu lietotāju no rezultātu saraksta, lai apskatītu viņa profilu.

Alternatīvi soļi:

* Ja meklēšanas rezultātos nav atrasts neviens lietotājs, sistēma paziņo par to un iedrošina mēģināt citus meklēšanas kritērijus.

**------------------------------------------------------------------------------------------------------------------------------------------**

**#3**

Programmatūras prasības ir kā pamati katram labam programmas projektam. Tās ir kā saraksts ar to, ko programmai jāspēj darīt, lai tā būtu noderīga lietotājiem. Piemēram, ja runājam par sociālo tīklu, tad prasības varētu būt tādas kā: var pievienot draugus, publicēt rakstus, komentēt utt.

Bet ne tikai "kas", bet arī "kā". Tas ir, kā programmai jāuzvedas, lai tā būtu ērta lietotājiem. Tāpēc ir svarīgi saprast, kā tā jāizskatās un jādarbojas - tas ir tās funkcionālās un nefunkcionālās prasības.

Lai šīs prasības precīzi definētu, ir nepieciešama laba sadarbība starp cilvēkiem, kas veido programmu, un tiem, kas to izmanto. Protams, ir dažādi veidi, kā to izdarīt - var izveidot prototipus, veikt pētījumus utt.

Un neaizmirstam par kvalitāti! Tas ir ļoti svarīgi, lai programma darbotos labi un būtu droša. Tāpēc pastāvīgi tiek veikti testi un pārbaudes, lai nodrošinātu, ka viss darbojas, kā vajag, un ka dati paliek drošībā.

**------------------------------------------------------------------------------------------------------------------------------------------**