**1. Problēmas izpēte un analīze**

**Izpētes metodes izvēle un pamatojums**

Projekta mērķis ir izveidot efektīvu un lietotājam draudzīgu darbinieku darba laika uzskaites sistēmu, kas nodrošina precīzu laika pierakstu, ziņu publicēšanu un administrācijas iespējas. Lai sasniegtu šo mērķi, tika veikta rūpīga izpēte, izmantojot dažādas metodes un pieejas.

Izpētei tika izmantotas sekojošas metodes:

* **Literatūras un tehniskās dokumentācijas izpēte** – lai saprastu labākās prakses darba laika uzskaites sistēmu izstrādē un izmantotu modernākās tehnoloģijas.
* **Esošo risinājumu analīze** – izpētot eksistējošas sistēmas un to funkcionalitāti, lai identificētu galvenās stiprās un vājās puses.
* **Intervijas ar mērķauditoriju** – lai identificētu prasības un uzlabojumu iespējas, balstoties uz lietotāju pieredzi un vajadzībām.
* **Prototipu testēšana** – agrīnās versijas funkcionalitātes pārbaude, lai uzlabotu lietojamību un pielāgotu interfeisu.
* **Anketēšana** – veicot datu vākšanu no potenciālajiem lietotājiem, lai noteiktu funkcionalitātes prioritātes.

**Izpētes procesa apraksts**

Projekta izstrāde tika veikta šādos posmos:

1. **Sistēmas prasību analīze** – noteiktas funkcijas un mērķauditorijas vajadzības, kā arī izstrādāti lietotāju scenāriji.
2. **Datu apstrāde un struktūras izveide** – izveidotas tabulas SQLite datu bāzē un definēti attiecību modeļi.
3. **Izstrāde un testēšana** – koda rakstīšana un atkļūdošana, kā arī sākotnējie integrācijas testi.
4. **Lietotāju testēšana un pielāgošana** – atgriezeniskās saites ievākšana un uzlabojumu ieviešana.
5. **Galīgā optimizācija un drošības pārbaudes** – sistēmas stabilitātes un veiktspējas nodrošināšana.

**Izpētes datu apkopojums**

Balstoties uz izpēti, tika secināts, ka galvenās prasības ir:

* Droša autentifikācija un autorizācija, izmantojot šifrēšanas metodes.
* Darba laika reģistrēšana un administrācija ar precīzu datu uzglabāšanu.
* Ziņu publicēšanas sistēma, kas ļauj administrācijai izplatīt informāciju.
* Administrācijas panelis lietotāju pārvaldīšanai, nodrošinot iespēju piešķirt un atņemt piekļuves tiesības.
* Responsīvs un lietotājam draudzīgs interfeiss, kas pielāgojas dažādām ierīcēm.

**2. Programmatūras prasību specifikācija**

**Mērķauditorijas izvēle un raksturojums**

Sistēmas mērķauditorija:

* Uzņēmumi un iestādes ar vajadzību pēc darbinieku darba laika uzskaites, īpaši attālinātā darba kontekstā.
* Administratori, kas vēlas pārvaldīt lietotājus un publicēt ziņas, lai informētu darbiniekus.
* Darbinieki, kas veic laika uzskaiti un skatās jaunākās ziņas, izmantojot lietotājam draudzīgu platformu.

**Programmatūras produkta un tā funkciju apraksts**

Sistēma piedāvā sekojošas funkcijas:

* **Lietotāja reģistrācija un pieteikšanās** – izmantojot Passport.js un bcrypt šifrēšanu.
* **Darba laika uzskaite** – clock-in un clock-out iespējas ar automātisku datu saglabāšanu.
* **Ziņu publicēšana** – administratori var pievienot un rediģēt ziņas, kas tiek parādītas visiem lietotājiem.
* **Administrācijas panelis** – lietotāju pārvaldība, admin lomu piešķiršana, tiesību maiņa un lietotāju apstiprināšana.
* **Piekļuves žurnāli** – iespēja sekot līdzi, kad lietotāji ir pieslēgušies un kādas darbības veikuši sistēmā.

**Programmatūras produkta skice**

Sistēma sastāv no:

1. **Servera puses (Node.js, Express, SQLite, bcrypt, Passport.js, dotenv)**
2. **Klienta puses (HTML, CSS, JavaScript, Bootstrap, AJAX)**
3. **Datu bāzes (SQLite) ar lietotāju, laika uzskaites un ziņu tabulām, iekļaujot audita žurnālus**

**3. Programmatūras izstrādes plāns**

1. **Prasību analīze un specifikācija** – 1.-2. nedēļa
2. **Sistēmas arhitektūras izstrāde un datu bāzes struktūras izveide** – 3. nedēļa
3. **Backend izstrāde un sākotnējā testēšana** – 4.-6. nedēļa
4. **Frontend izstrāde, responsīva dizaina pielāgošana** – 7.-8. nedēļa
5. **Integrācija un drošības testi** – 9.-10. nedēļa
6. **Pilnvērtīga testēšana, lietotāju atsauksmju ievākšana** – 11. nedēļa
7. **Dokumentācijas sagatavošana un nodošana** – 12. nedēļa

**4. Atkļūdošanas un akcepttestēšanas pārskats**

**Testēšanas metodes:**

* Vienības testēšana (unit tests), lai nodrošinātu atsevišķu moduļu korektu darbību.
* Integrācijas testēšana, lai pārbaudītu komponentu mijiedarbību.
* Lietotāju testēšana, balstoties uz reālās darbības scenārijiem.
* Drošības testēšana, īpaši koncentrējoties uz autentifikācijas un datu aizsardzību.

**Būtiskākie labojumi un optimizācijas:**

* Lietotāju autentifikācijas un paroļu drošība.
* Datu bāzes vaicājumu optimizācija un indeksēšana.
* Sistēmas darbības ātruma uzlabošana un lietojamības testēšana.

Man bieži vien bija problēmas ar database sistēmu, kur bija jāregistrē lietotāji. Arī news sistēma aizņēma daudzus mēģinājumus, līdz kamēr sanāca.

**6. Piemērotās licences pamatojums**

Sistēma tiek izplatīta ar **MIT licenci**, kas nodrošina atļauju brīvi izmantot, modificēt un izplatīt kodu, vienlaikus saglabājot autortiesību paziņojumus.