24.02.2020

training manager

**Technology stack:**

SQL-Server: MS SQL Server (preferred) (Mysql - as alternative);

web server: internet information server (apache - as alternative);

Server development engine: asp.net core MVC (C#) (php - as alternative);

Middleware/services: web api (or any restful enable WEB SERVICE endpoints - as alternative);

front end: html, css (bootstrap), typescript/javascript, jquery, angular OR knockout-js;

**Terminology:**

* Repository – is an abstraction under O/R mapping database operations;
* Unit of work – is a façade which encapsulates all repositories as one central place to access from the code base;
* Dependency injection – the set of actions to avoid the tight connection between architecture components;
* O/R Mapping – Object relational mapping, a way to map database tables into the classes with some attributes;
* Architecture – (in the project context) a set of actions and development activities to avoid complexity, to keep up coherence during the development process, to avoid cyclomatic complexity between the code and components as the code base grows;
* Backend - (in the project context) a set of development activities which are not able to have any UI-presentation, usually it’s data related tasks;
* Consolidation – review tasks to reach an agreement and align requirements between customer and developer points of view;
* Frontend - (in the project context) a set of development activities which always have user interface;
* Middleware - (in the project context) a set of development activities which take a responsibility to be a connection point between data and presentation (let’s say as it would be in the middle);
* HTTP controller – the middleware code to handle HTTP requests/ responses;
* Server - (in the project context) a set of development activities related to the server context;
* Unit test – a code coverage activity to make sure that the previously developed components haven’t been broken in the current development phase;
* UI test – automation and/or manual testing of presentation screens and expected results;
* CRUD – expected activities with data: **C**reate, **R**etrieve, **U**pdate, **D**elete;
* WEB API – Web service stack with RESTful support for the main HTTP-methods: GET, POST, PATCH, HEAD, DELETE, PUT; extremely useful for generic clients such as mobile devices, web clients, desktop clients etc.

| Task | ESTIMATION | TYPE OF WORK |
| --- | --- | --- |
| **BEGINNING ACTIVITIES** |  | **-------------------** |
| Checking out the database scheme for the 3rd form normalization (modifying if needed) | 4h | Architecture |
| Creating raw SQL-script for database entities with parent/child relationship commands (Initial DDL-script) | 12h | Backend |
| Seeding the sample data for tables | 4h | Backend |
| Creating O/R mapping classes and initial debug/release configuration | 6h | Backend |
| Creating initial Repository-class and Unit of work Façade for database operations | 4h | Architecture |
| Defining the primary auth-roles for code first authority system: Regular user->Trainer->Admin. | 2h | Backend |
| UI Web prototype with navigation scheme | 20h | Architecture |
| Activity/Sequence UML-diagram for Admin-role | 5h | Architecture |
| Activity/Sequence UML-diagram for Trainer-role | 3h | Architecture |
| Activity/Sequence UML-diagram for Regular user role | 6h | Architecture |
| Review intermediate results with Felix | 2h | Consolidation |
| **DEVELOPMENT ACTIVITIES** |  | **-------------------** |
| Layout implementation for header according to UI-prototype | 3h | Frontend |
| Layout implementation for middle according to UI-prototype | 5h | Frontend |
| Layout implementation for footer according to UI-prototype | 2h | Frontend |
| Layout implementation for menu/navigation according to UI-prototype | 4h | Frontend |
| Developing index screen for Admin | 2h | Frontend |
| Developing View trainings screen for Admin | 4h | Frontend |
| Developing Add training screen for Admin | 4h | Frontend |
| Developing Edit/Update training screen for Admin | 3h | Frontend |
| Developing HTTP-controller for admin screens activities | 6h | Middleware |
| Adding delete HTTP-controller action | 2h | Middleware |
| Linking admin screens with HTTP-controller | 4h | Server |
| Developing Unit test coverage for Admin HTTP-controller | 4h | Unit test |
| Testing Admin screens UI | 2h | UI Test |
| Creating Notification interface | 3h | Architecture |
| Creating On Adding trigger to subscribe users | 4h | Server |
| Implementing notification interface, linking dependency injected implementation with HTTP-controller | 6h | Server |
| Developing Report screen for Admin | 3h | Frontend |
| Adding report action to Admin HTTP-controller | 3h | Middleware |
| Adding 2 “export as” actions to Admin HTTP-controller – for excel and html formats | 12h | Server |
| Linking Report screen with actions | 3h | Frontend |
| Adding filter support to Admin HTTP-controller | 6h | Middleware |
| Adding UI for filter strategies – mapping them to URL route system | 6h | Frontend |
| Linking Report screen with filter-actions | 3h | Frontend |
| Adding support for managing work packages via HTTP controller instance | 10h | Middleware |
| Adding support for managing GroupWise appointments via HTTP controller instance | 10h | Middleware |
| Creating CRUD screens for training instances | 8h | Frontend |
| Adding actions to HTTP-controller for CRUD training instances | 8h | Middleware |
| Linking admin screens with external API’s (GroupWise, ProjectXT) | 10h | Frontend |
| Review intermediate results with Felix | 4h | Consolidation |
| Developing CRUD Training instance views | 12h | Frontend |
| Developing HTTP controller for Trainer endpoint activities | 6h | Middleware |
| Developing View participations screen | 3h | Frontend |
| Adding workflow action for Confirm->Request->notification for trainer participation screen | 6h | Middleware |
| Linking Trainer screens with HTTP controller | 3h | Frontend |
| Unit test coverage | 4h | Unit test |
| UI testing of recently created screens | 3h | UI Test |
| Developing CRUD screens for training ratings | 10h | Frontend |
| Creating HTTP controller for viewing training ratings with filtering support | 8h | Middleware |
| Link training ratings with HTTP controller | 4h | Server |
| Developing screen for viewing training requests by Admin | 3h | Frontend |
| Adding actions to admin HTTP controller for viewing training requests with filtering support | 4h | Middleware |
| Altering database-index for full text search support | 2h | Backend |
| Adding UI controls for text search | 4h | Frontend |
| Link training requests screen with HTTP controller | 2h | Server |
| Developing training list view for regular user | 3h | Frontend |
| Creating HTTP controller for regular user actions | 3h | Middleware |
| Adding UI controls for filtering training instances for regular user | 5h | Frontend |
| Linking training list screen with HTTP controller | 2h | Server |
| Adding filter strategies to regular user HTTP controller for viewing training instances | 6h | Middleware |
| Creating screen for viewing training instances for regular user | 3h | Frontend |
| Creating HTTP controller for regular user’s training instance with filtering support | 4h | Middleware |
| Linking training list screen with HTTP controller | 2h | Server |
| Adding UI controls for filtering training list | 5h | Frontend |
| Synchronize UI-filters with HTTP controller’s actions | 8h | Server |
| Unit test coverage for recently developed controller actions | 6h | Unit test |
| UI testing for training list views | 2h | UI Test |
| Review intermediate results with Felix | 3h | Consolidation |
| Developing screen for viewing users by regular user | 3h | Frontend |
| Adding UI controls for filtering support | 3h | Frontend |
| Developing HTTP controller for list user actions by regular user, with filtering support | 6h | Middleware |
| Linking screen for viewing users with HTTP controller | 2h | Server |
| Adding support for participation request with notification trigger | 4h | Server |
| Developing wizard screen for participation request by regular user | 7h | Frontend |
| Adding Rate training workflow activities with notification trigger | 5h | Server |
| Developing wizard screen for rate training workflow | 5h | Frontend |
| Adding support for suggest training creation with notification trigger | 4h | Server |
| Developing wizard screen for suggest training view | 4h | Frontend |
| Developing PDF/RTF template for generating certificates with signing | 20h | Backend |
| Adding support for generating certificate by regular user | 6h | Server |
| Creating HTTP controller for certificate generator support | 6h | Middleware |
| Developing view for generating certificate | 4h | Frontend |
| Linking view with HTTP controller | 3h | Server |
| Tuning notification template for e-mail | 4h | Server |
| Coverage unit test for recently created controller actions | 8h | Unit test |
| UI Test for recently created screens | 3h | UI Test |
| Review intermediate results with Felix | 3h | Consolidation |
| **ACCOUNT MANAGEMENT SCREENS** |  | **-------------------** |
| Creating user manager infrastructure for 3 roles | 8h | Backend |
| Creating register view | 4h | Frontend |
| Creating login view | 2h | Frontend |
| Creating edit user credentials view | 6h | Frontend |
| Developing HTTP controller for register view | 4h | Middleware |
| Developing HTTP controller for login view | 3h | Middleware |
| Developing HTTP controller for edit user credentials | 6h | Middleware |
| Tuning claims and auth-filtering depending on user role rights | 4h | Server |
| Developing CRUD-screens for user rights – for admin only | 8h | Frontend |
| Developing HTTP controller for user right manipulation activities | 6h | Middleware |
| Linking CRUD activities with HTTP controller | 2h | Server |
| Synchronizing existing active directory entries | 5h | Architecture |
| Coverage unit test for recently created controller actions | 3h | Unit test |
| UI Test for recently created screens | 2h | UI Test |
| Review intermediate results with Felix | 2h | Consolidation |
| **IMPROVEMENTS AFTER Consolidation** |  | **-------------------** |
| UI views improvements | 20h | Frontend |
| Refactoring HTTP controllers to keep up the low coupling, high cohesion for domain layers architecture | 20h | Architecture |
| **DEPLOYMENT ACTIVITIES** |  | **-------------------** |
| Aligning with Server PC requirements | 3h | Deployment |
| Administrative activities for database server | 6h | Deployment |
| Aligning with IIS/Apache requirements | 4h | Deployment |
| Developing web deployment package for release | 5h | Deployment |
| Deploying release and investigating fallbacks | 4h | Deployment |
| Looking through HTTP speed performance and 4xx, 5xx HTTP-errors | 4h | Deployment |