­7.1P – Securing Your Enterprise Application

Task 4 – Analysis Task

1. **In the context of the employee’s CRUD operations on their own record, why the system does not allow employee to perform the “C” and “D” operations? Justify your answers.**

The system doesn’t allow an employee to perform a create or delete option because it would allow them to change details that would otherwise be immutable by simply deleting their record and recreating

1. **In the context of the employee’s Review operation, what information can be reviewed? Justify your answer.**

Empid, name, phone, address, email, bankaccountid, & salary.   
These fields should be reviewable as they all directly relate to the employee.

Password is not shown as it could be seen by other employees in the same physical location.

Appgroup & active are not shown as they aren’t relevant to the employee and partially reveal how the system is secured.

1. **In the context of employee’s Review operation (reviewing their own detail), the company decided to implement a DTO which excludes the password being sent to the client. Why password is excluded? Do you think that this is a good practice? Why or Why not? If not, propose an alternative and justify your choice.**

The password is excluded because its unnecessary to display to the user, and risks someone in the same physical location seeing it.

I think this is a good practice because it removes a possible security risk.

1. **In the context of the employee’s Update operation,**
   1. **What information can be updated? Are these the same as those in 4.2 and 4.3 above? Why or Why not? Justify your answer.**

Name, Phone, Address, email, password, bankaccountid.

These can be changed because they directly relate to either the employee personally, or to their use of the application.

The password can be changed but not viewed, as its expected the employee memorises their password.

* 1. **What information cannot be updated? How would you avoid these data being updated by the employee “accidentally”?**

Empid, appgroup, salary, active status.

To avoid updates there would be a separate screen with only the fields we want the employee to be able to update.

1. **In the context of the employee’s Update operation, where should the actual change of the employee’s information occur? Do you think this is a good practice? Why or Why not? Justify your answer.**

On a separate employee only update page. I think this is good practice as it allows full control over what fields can be updated, and removes the possibility of bugs where the user can update fields only an admin should be able to that may occur using a shared ui.

1. **In the context of employee’s Update operation, the company decided to first display the details of a particular employee (if such employee exists after searching through the database via the employee’s id) in the web browser so that the employee could enter the required information. Should the existing password be**
   1. **sent and displayed to the client?**
   2. **sent to the client but not displayed?**
   3. **not sent?**

**What is your choice? Why or why not? Justify your answer. If your answer is (3), how would you implement the feature that allows the employee to change their own password?**

C. There’s no reason for the client to have a copy of the users current password. The managed bean on the server would contain it, and password updates would occur by having the user submit their old and new password and the managed bean would only allow it if the submitted old password matches the actual old password.

1. **In the context of deleting employee’s record, the company choose to accept the employee id as the input and then remove the employee record by setting the field “active” to false instead of removing the record from the database. Do you think that this is a good practice? Why or Why not? If not, propose an alternative and justify your choice.**

This is more of a business choice as keeping a list of inactive users allows for collecting historical data about the business. This seems like an ok practice as I can’t think of any way this could be abused.

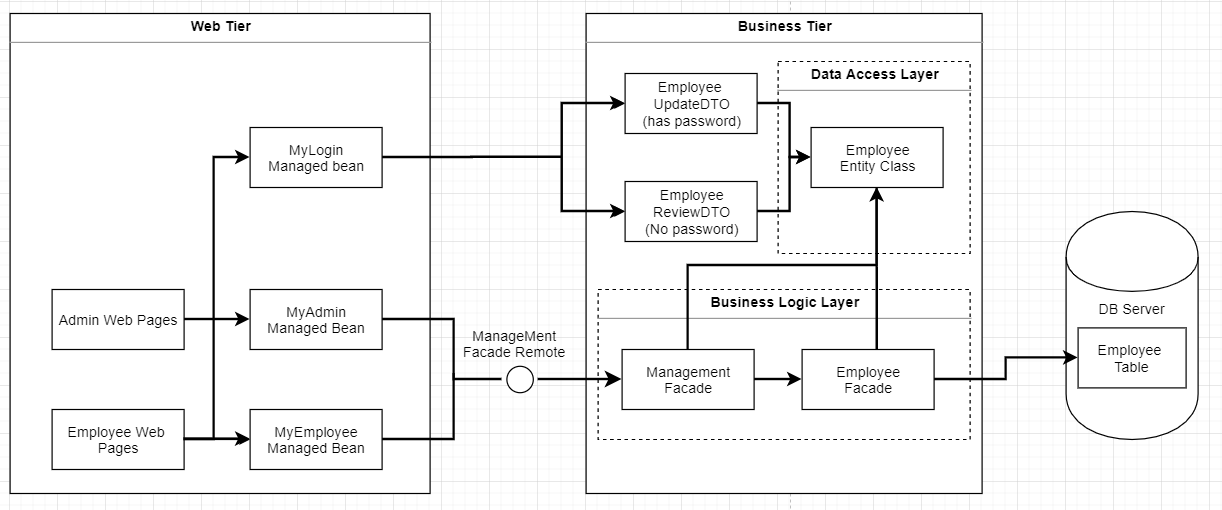
1. **After reviewing the features provided by the application “ED-Secure”, do you think that the application as is can provide the features listed in the Case Study Section above? Are there any deficiencies? Why and why not? What changes would you suggest to address all features listed?**

The application needs the following changes:

* Separate user only jsf pages for employees to view and update their data
* New DTO class that omits password

Task 5

* 1. **Uml diagram**



**b. Component Descriptions**

* Admin Web Pages

The set of jsf webpages that provide the UI for the admin tasks to be completed in the application

* Employee web pages

JSF web pages in which the employee can navigate to review and update their details.

* Employee UpdateDTO

DTO specific for transferring updated details to the managed bean. This contains the password fields for updating user password

* Employee Review DTO

DTO for transferring employee review values from managed beans to the client. Does not contain a password field

* MyAdmin Managed Bean

Holds session data & provides functionality specific to admin pages.

* MyEmployee Managed Bean

Holds session data & provides functionality specific to employee pages. Notably different to the admin managed bean as it will lack methods for adding & deleting users, and will only retrieve data for the current user.

* Mylogin Managed Bean

Provides functionality for logging the user out of their login session.

* Management Façade

Provides data access methods to managed beans.

* Employee Façade

Provides data access methods for updating employees in the database.