

Design Document	Rev. 4
HR Employee Tracking Application	Page 1 of 22

HR Employee Tracking Application

Software Design Document

Revision History

Name	Role	Purpose	Date	Document Status
Zach Comstock	Documentation Engineer	Initial Creation	10/5/17	Draft
Zach Comstock	Documentation Engineer	Wireframes complete	10/10/17	Draft
Zach Comstock	Documentation Engineer	System Design and Database Design Complete	10/17/17	Draft
Zach Comstock	Documentation Engineer	Revisions to Database and System Design	11/10/17	Complete

Table of Contents

1. Introduction	3
1.1 Purpose	3
1.2 Scope	3
1.3 Overview	4
2. Use Cases	4
2.1 Description	4
2.2 Written Use Cases	5
2.3 Use Case Flows	9
3. Design Overview	15
3.1 Introduction	15
3.2 Functional Decomposition	15
4. Database Design	16
4.1 ER Diagram	16
5. Wireframes	17
5.1 Wireframe Sketches	17
5.2 Finalized Wireframes	19

Design Document	Rev. 4
HR Employee Tracking Application	Page 3 of 22

1. Introduction

1.1 Purpose

The purpose of this document is to clarify the design details about the system we intend to build, as well as how the user will view and interact with the system; this entails detailing which cases shall be handled by the system and what interactions are expected of the user. The system design needs to accommodate all flows for all use-cases detailed in this document. The design overview will cover the technology and system architecture we will be using for this project; this will be useful for readers to gain a better idea of how the data will move through the system. Data objects and the database design will be detailed in section four of this document. Lastly, in order to account for user interaction and user experience, we have some simple prototypes and wireframes towards the end of this document.

This document will likely go through various changes throughout the development process. As a developer, if you find any issues or have questions/concerns with the documentation then please notify the team. If changes need to be made, a documentation engineer will made the necessary changes and inform you upon completion.

1.2 Scope

The ultimate goal of this project, as it is understood, is to create a software application that is able to track and monitor an employee's current standing in the company. The project shall hereby be referred to as the "HR Employee Tracking Application". The target user group for the HR Employee Tracking Application is expected to primary be Human Resources. However, the application will also be used by managers and general employees, with restricted functionality of course. The HR Employee Tracking Application is intended to assist Human Resources with the management and tracking of employees' and managers' productivity and employment information vacation and sick hours. Human Resources shall also be able to use the HR Employee Tracking Application to approve or employees' and managers' requests, such as vacation, raise, or promotion requests.

The application is meant to be a web based application containing a frontend to handle the views, a backend to process the logic, and a database to store the data. The database should handle modifications to the information (add, read, edit, delete) stored within based on the permissions granted to each specific user.

In order to create a fully functioning HR Employee Tracking Application, the project must include the following features per each user group:

User Group	Functionality Needs
Employees	Employees should be able to view all of this information and be able to file a discrepancy complaint if something is inconsistent with what they know. HR Should be able to send this complaint to the manager for them to review. Employees should also be able to request vacation and it should show that the vacation days are subtracted from the amount they accumulated over the course of their work. In the same way, employees should be able to request sick leave. The manager should get a notification that their employee is requesting sick leave and should be able to easily see whether or not the request can be

Design Document	Rev. 4
HR Employee Tracking Application	Page 4 of 22

	confirmed or denied. Employees should be able to also quit their job whenever they so choose by putting in a two weeks notice.
Managers	The managers should be able to write reviews on their employees, approve or deny sick leave & vacations, and see whether or not an employee is in good standing with the company. The manager should also be able to give raises to an employee, which should be cleared by HR before being set into place.
Human Resources	The HR Department should be able to terminate employees, but still keep their records on file. They should be able to approve of vacations that exceed the number set by the company and a negative result should go into the manager's file if that occurs. HR should be able to send a manager a warning about potentially failing employees. HR should also be able to accept or deny increased in salary and decreases in salary.

This project shall provide a convenience for Employees and Human Resources alike, that we believe will be used heavily. The needs to be simple and easy to use in order to ensure fast turnaround, so that employees will be able to get back to more pressing tasks which will in turn increase productivity. The system design is also meant to be expandable incase new additions or features need to be added to the application in the future.

1.3 Overview

Being that the HR Employee Tracking Application will be a primarily data-driven web application, there will likely be no need for object-oriented programming. Therefore, there will be no object models or UML class diagrams.

The primary focus of this document will be to identify how the user will interact with the application in order to submit a query for data processing. You will find that most, if not all, user interactions can be boiled down to a single request to add/edit/read/delete data from the SQL Database. Understanding how users wish to interact with the systems' front-end in order to submit these requests will be essential for having a good, user friendly, design.

Therefore, the bulk of this document will be cover users interaction and users experience with the application (How the users should interact with the front-end and how the front-end should look). Secondly, this document will cover the overall system design, identifying how data will flow from the back-end to the front-end of the system. Lastly, being that this is a data centered application, we will explicitly identify all the tables that will be needed in the SQL Database.

2. Use Cases

2.1 Description

It is our goal to keep the following use-cases as simple as possible, primarily identifying the user's interaction with the system. As a programmer, you must ensure that each use-case flow can be completed as it is described, while still maintaining a sleek user-friendly design that closely resembles the wireframe designs shown at the end of this document. Provide all functionality, without sacrificing usability.

Design Document	Rev. 4
HR Employee Tracking Application	Page 5 of 22

In the scenario that the functionality cannot be provided without compromising the initial design, report it to the rest of the team and discuss a potential work around or solution. Be sure not to leave out any use-case flow, even if the design must be changed slightly.

In order to ensure correct implementation, critical and complex functionalities have been detailed in written use-cases. Less complex functionalities are show in a simplified version of a use-case diagram, a use-case flow.

The following use-cases will involve these actors: General Employee, Manager, Human Resources, Human Resources Manager, SQL Database. In the situation that there are more than 1 of a type of actor (2 employees for example), the second actor will be denoted a '2' after its name.

If there is any ambiguity in these use-cases, do not hesitate to contact the documentation engineers for clarification. He/she will correct/revise any issues in the document and notify you upon its completion.

2.2 Written Use Cases

Use Case Name:	Request Time-Off		
Use Case ID:	000001		
Date Created:	11-16-2017	Date Last Updated:	n/a
Author:	Zach Comstock	Last Updated By:	n/a

Actors:	General Employee, Manager, SQL Database		
Summary:	A General Employee wishes to request time-off (either sick-leave or vacation). The manager and human resources will later have the option to either approve or deny the request.		
Trigger:	General Employee navigates to "request time-off" view, fills in information, and presses the "submit" button		
Assumptions:	<ul style="list-style-type: none"> The General Employee will still be employed by the company during the time period that the employee requests off. The General Employee is not the HR manager. The General Employee does not request an unrealistic amount of time off (greater than 1 month) 		
Pre-conditions:	<ul style="list-style-type: none"> General Employee must have available time-off (either vacation time or sick leave) General Employee must have a manager associated with his/her account. 		
Post-conditions:	<ul style="list-style-type: none"> A request for time-off has been added to the SQL Database. The General Employee's Manager has been notified of the request. 		
Normal Flow:	<ol style="list-style-type: none"> General Employee is currently on the "request time-off" view. General Employee selects the type of time-off request he/she wishes to request General Employee chooses the start and end date for the desired time-off. General Employee fills out the reason for the request if applicable. General Employee presses the "Submit" button. The Request is added to the SQL Database The General Employee's Manager is sent a notification, informing him/her about the new time-off request 		
Alternative Flows:	<ol style="list-style-type: none"> General Employee is currently on the "request time-off" view. General Employee selects the type of time-off request he/she wishes to request General Employee chooses the start and end date for the desired time-off. General Employee fills out the reason for the request if applicable. General Employee presses the "Submit" button. 		

Design Document	Rev. 4
HR Employee Tracking Application	Page 6 of 22

	6. General Employee has an insufficient amount of time-off for the requested leave type. 7. The General Employee is notified that the request has failed and that he/she has an insufficient amount of time-off for the requested leave type.
Exceptions:	<ul style="list-style-type: none"> General Employee did not fill out all fields before pressing “Submit” button General Employee does not have a Manager associated with his/her account General Employee has insufficient time-off in his/her account

Use Case Name:	Approve Time-Off		
Use Case ID:	000002		
Date Created:	11-16-2017	Date Last Updated:	n/a
Author:	Zach Comstock	Last Updated By:	n/a

Actors:	General Employee, Manager, Human Resources, SQL Database		
Summary:	A Manager has received a request for time-off (either sick-leave or vacation) from one of his/her General Employees. The manager wants to either approve the request (sent to Human Resources) or deny it.		
Trigger:	Manager navigates to “Messages” view, to see a list of incoming time-off requests.		
Assumptions:	<ul style="list-style-type: none"> The General Employee will still be employed by the company during the time period that the employee requested off. The General Employee is not the HR manager. The Manager is the manager of the General Employee 		
Pre-conditions:	<ul style="list-style-type: none"> Manager has pending time-off requests. 		
Post-conditions:	<ul style="list-style-type: none"> The Human Resources has been notified of the requests approval. 		
Normal Flow:	1. Manager is currently on the “Messages” view. 2. Manager selects “Approve” button. 3. The Request is sent to Human Resources for final approval.		
Alternative Flows:	1. Manager is currently on the “Messages” view. 2. Manager selects “Deny” button. 3. Employee is notified that his/her request has been denied.		

Use Case Name:	File Complaint		
Use Case ID:	000003		
Date Created:	11-16-2017	Date Last Updated:	n/a
Author:	Zach Comstock	Last Updated By:	n/a

Actors:	General Employee, General Employee 2, SQL Database		
Summary:	A General Employee wishes to file a complaint about General Employee 2.		
Trigger:	General Employee navigates to “File Complaint” view.		
Assumptions:	<ul style="list-style-type: none"> The General Employee is logged-in and has a currently active session. 		
Pre-conditions:	<ul style="list-style-type: none"> General Employee 2 exists in the SQL Database 		
Post-conditions:	<ul style="list-style-type: none"> General Employee 2 has been notified of the complaint. Complaint record is added to the SQL Database 		
Normal Flow:	1. General Employee current on “File Complaint” view. 2. General Employee types General Employee 2’s name into the field. 3. General Employee types the contents of the complaint into the field.		

Design Document	Rev. 4
HR Employee Tracking Application	Page 7 of 22

	<ol style="list-style-type: none"> General Employee selects the “Submit” button. Record of the complaint is added to the SQL Database. General Employee 2 is notified of the complaint.
Alternative Flows:	<ol style="list-style-type: none"> General Employee current on “File Complaint” view. General Employee types General Employee 2’s name into the field. General Employee 2 is no longer employed by the company. General Employee is notified of this.
Exceptions:	<ul style="list-style-type: none"> General Employee selects “Submit” before filling out all fields.

Use Case Name:	HR Approve New Position		
Use Case ID:	000004		
Date Created:	11-16-2017	Date Last Updated:	n/a
Author:	Zach Comstock	Last Updated By:	n/a

Actors:	General Employee, Manager, Human Resources, Human Resources Manager, SQL Database
Summary:	A Human Resources Employee has received a request for an employee promotion (new position) from a General Employee or Manager. The Human Resources Employee wants to either approve the request or deny it.
Trigger:	Human Resources Employee navigates to “Messages” view, to see a list of incoming requests.
Assumptions:	<ul style="list-style-type: none"> The General Employee will still be employed by the company during the time period that the employee requested off. The General Employee is not the HR manager. The Employee is not currently holding the same position as the Human Resources Employee
Pre-conditions:	<ul style="list-style-type: none"> Human Resources Employee has pending time-off requests.
Post-conditions:	<ul style="list-style-type: none"> The new position request has been approved. The new position request has been denied. Employee and Manager are informed of the decision.
Normal Flow:	<ol style="list-style-type: none"> Human Resources Employee is currently on the “Messages” view. Human Resources Employee selects “Approve” button. Record of Employee’s position is modified to reflect the newly approved position and salary Employee and Manager are notified of the acceptance.
Alternative Flows:	<ol style="list-style-type: none"> Human Resources Employee is currently on the “Messages” view. Human Resources Employee fills out details for refusal Human Resources Employee selects “Deny” button. Employee and Manager are notified of the rejection.
Exceptions:	<ul style="list-style-type: none"> Human Resources Employee selects “Approve” or “Deny” before filling out all fields.

Use Case Name:	Terminate Employee		
Use Case ID:	000005		
Date Created:	11-16-2017	Date Last Updated:	n/a
Author:	Zach Comstock	Last Updated By:	n/a

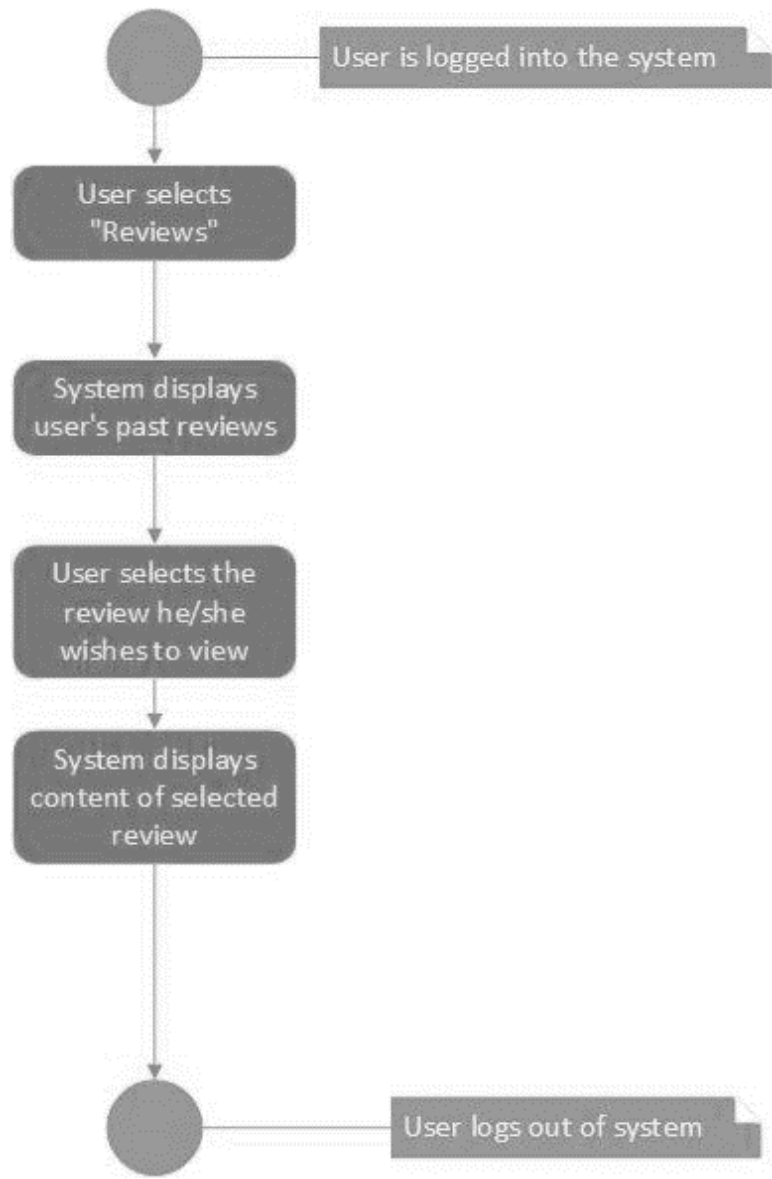
Actors:	General Employee, Manager, Human Resources, SQL Database
---------	--

Design Document	Rev. 4
HR Employee Tracking Application	Page 8 of 22

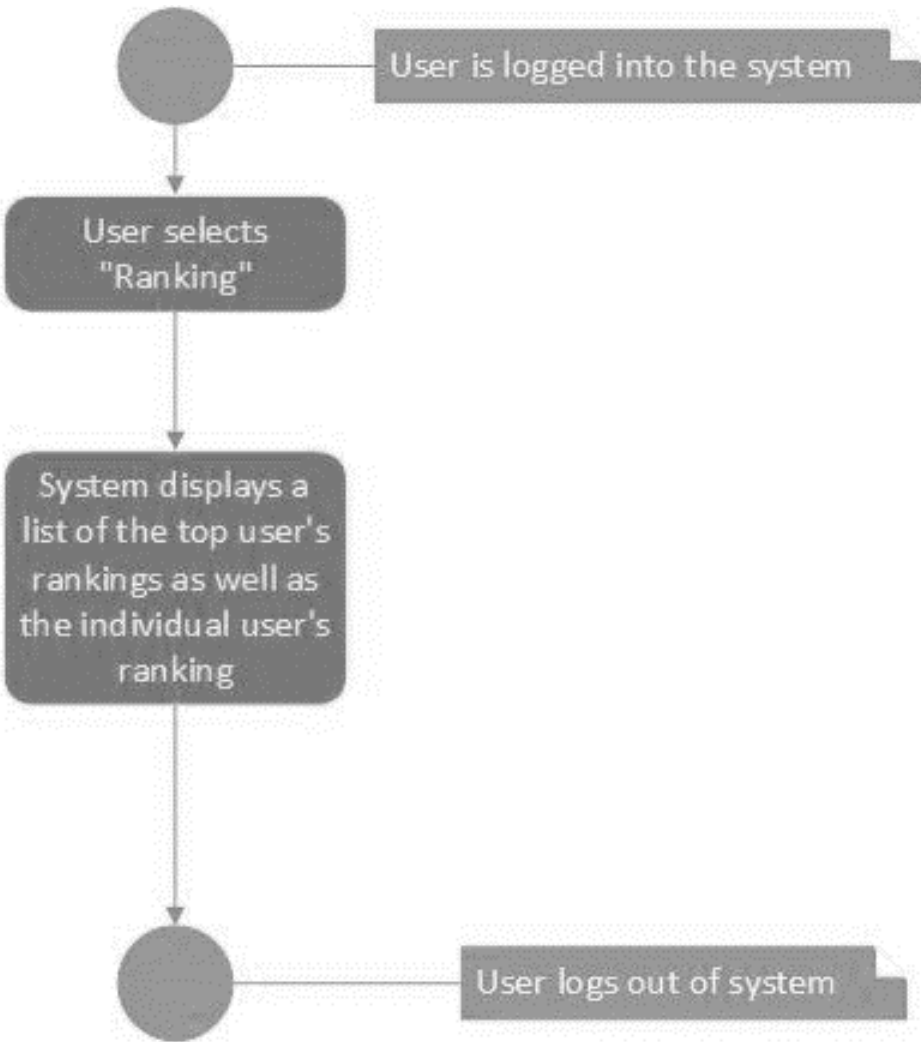
Summary:	A Human Resources Employee need to terminate an employee.
Trigger:	Human Resources Employee navigates to “Employees” view.
Assumptions:	<ul style="list-style-type: none"> The Human Resources Employee has sufficient permission to terminate the General Employee.
Pre-conditions:	<ul style="list-style-type: none"> The General Employee to be terminated holds a lower position than the Human Resources Employee who is doing the termination.
Post-conditions:	<ul style="list-style-type: none"> The General Employee and his/her manager are notified of his/her termination and the termination date The SQL Database is updated to reflect that the General Employee’s last day is on the termination date.
Normal Flow:	<ol style="list-style-type: none"> Human Resources Employee is currently on the “Employee” view. Human Resources Employee navigated to the General Employee to be fired. Human Resources Employee fills out field for reason of termination. Human Resources Employee selects “Terminate Employee” Employee and Manager are notified of the Employee’s termination.
Alternative Flows:	<ol style="list-style-type: none"> Human Resources Employee is currently on the “Employee” view. Human Resources Employee navigated to the General Employee to be fired. Human Resources Employee neglects to fill out field for reason of termination. Human Resources Employee selects “Terminate Employee” Human Resources Employee is notified that the request did no go through and is advised to fill out the blank field. Human Resources Employee fills out field for reason of termination. Human Resources Employee selects “Terminate Employee” Employee and Manager are notified of the Employee’s termination.

2.3 Use Case Flows

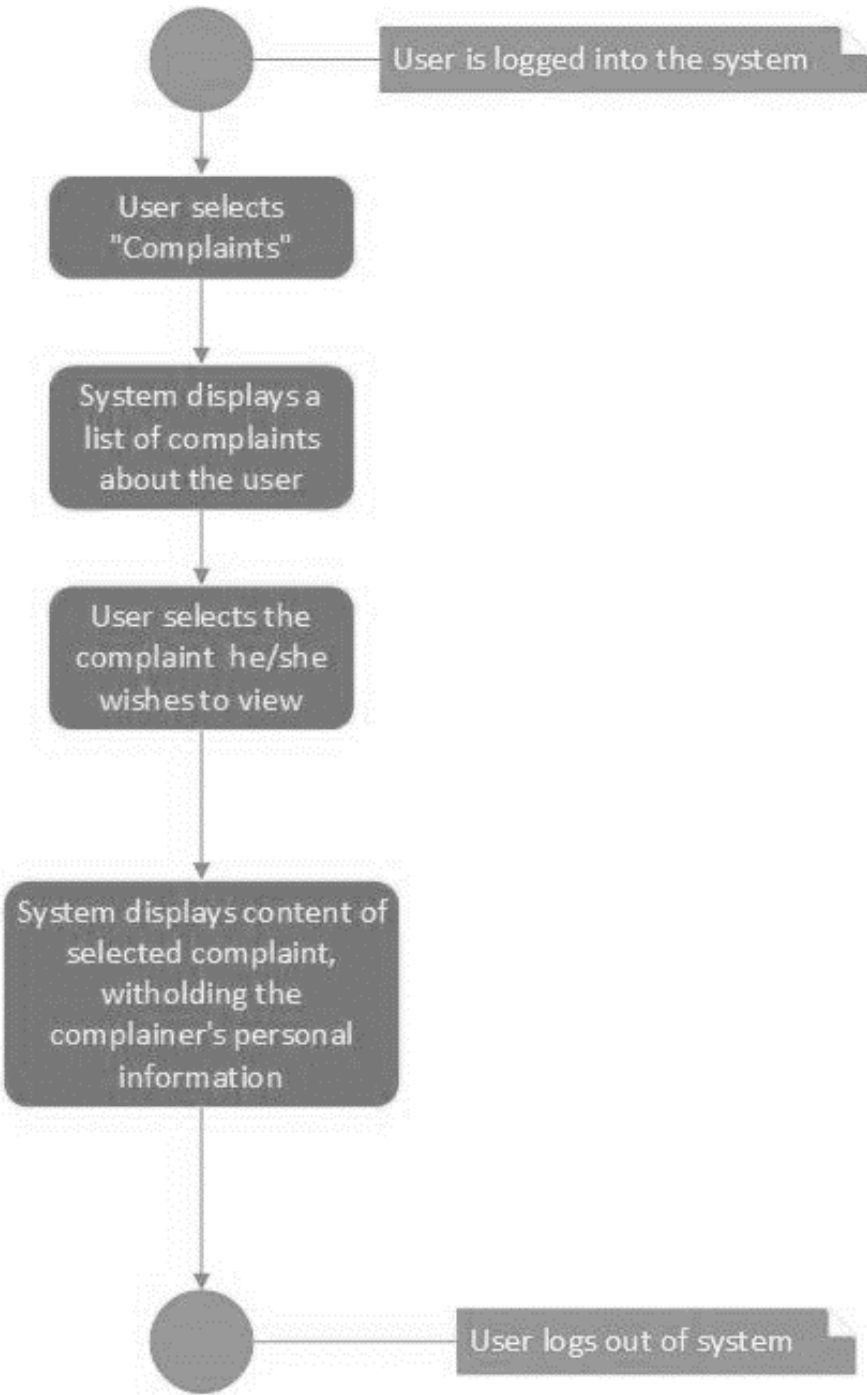
View Review



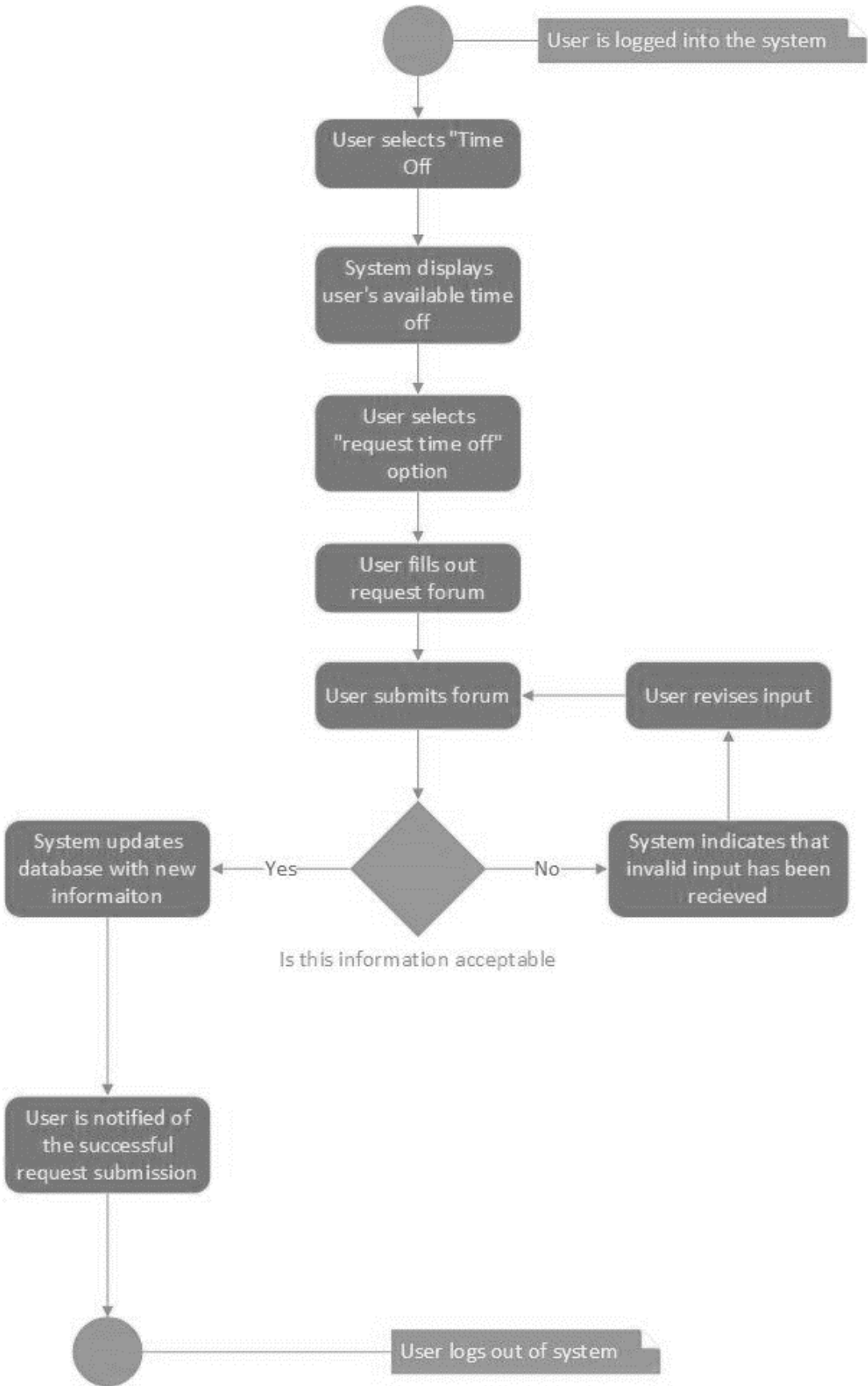
View Ranking



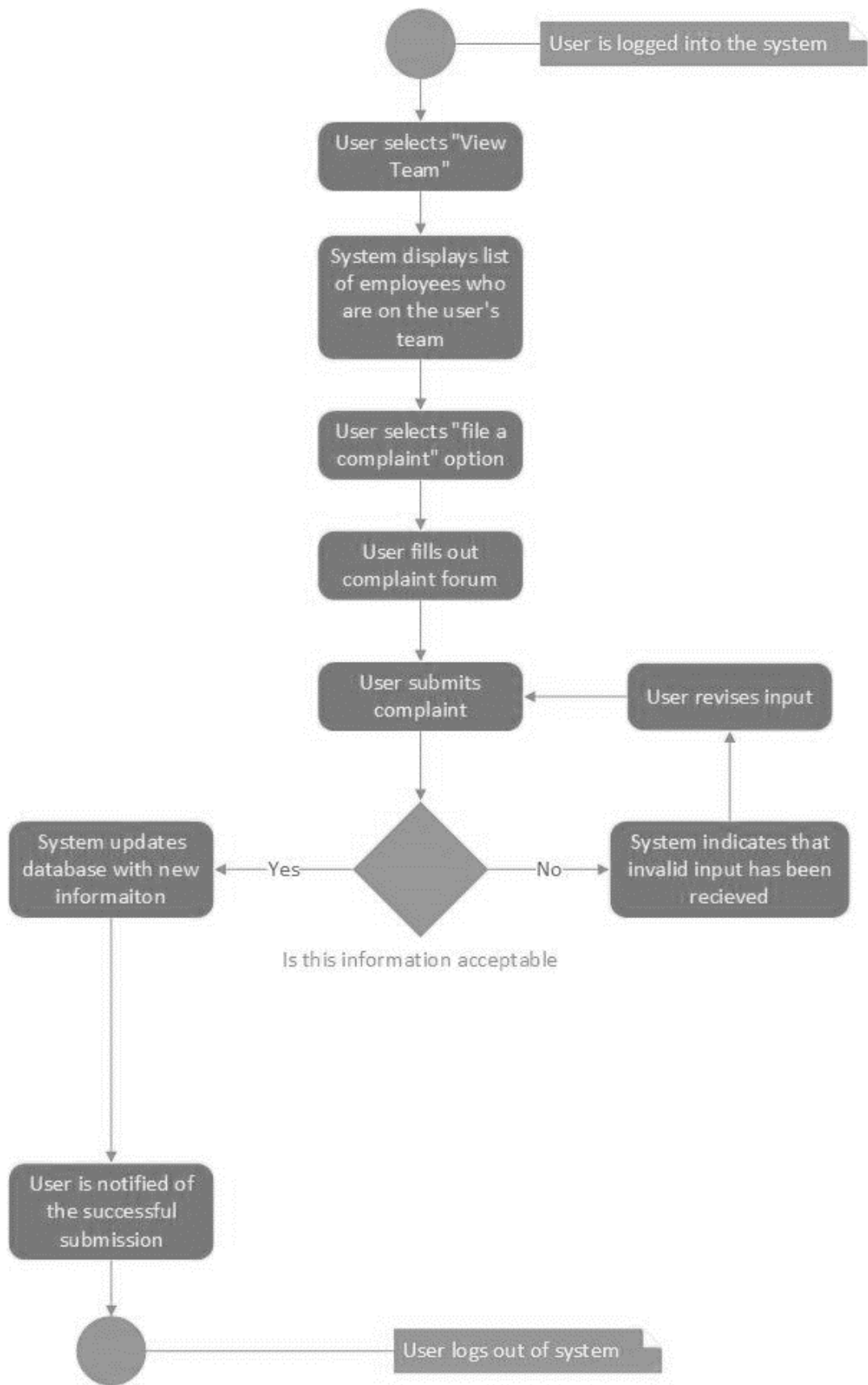
View Complaint



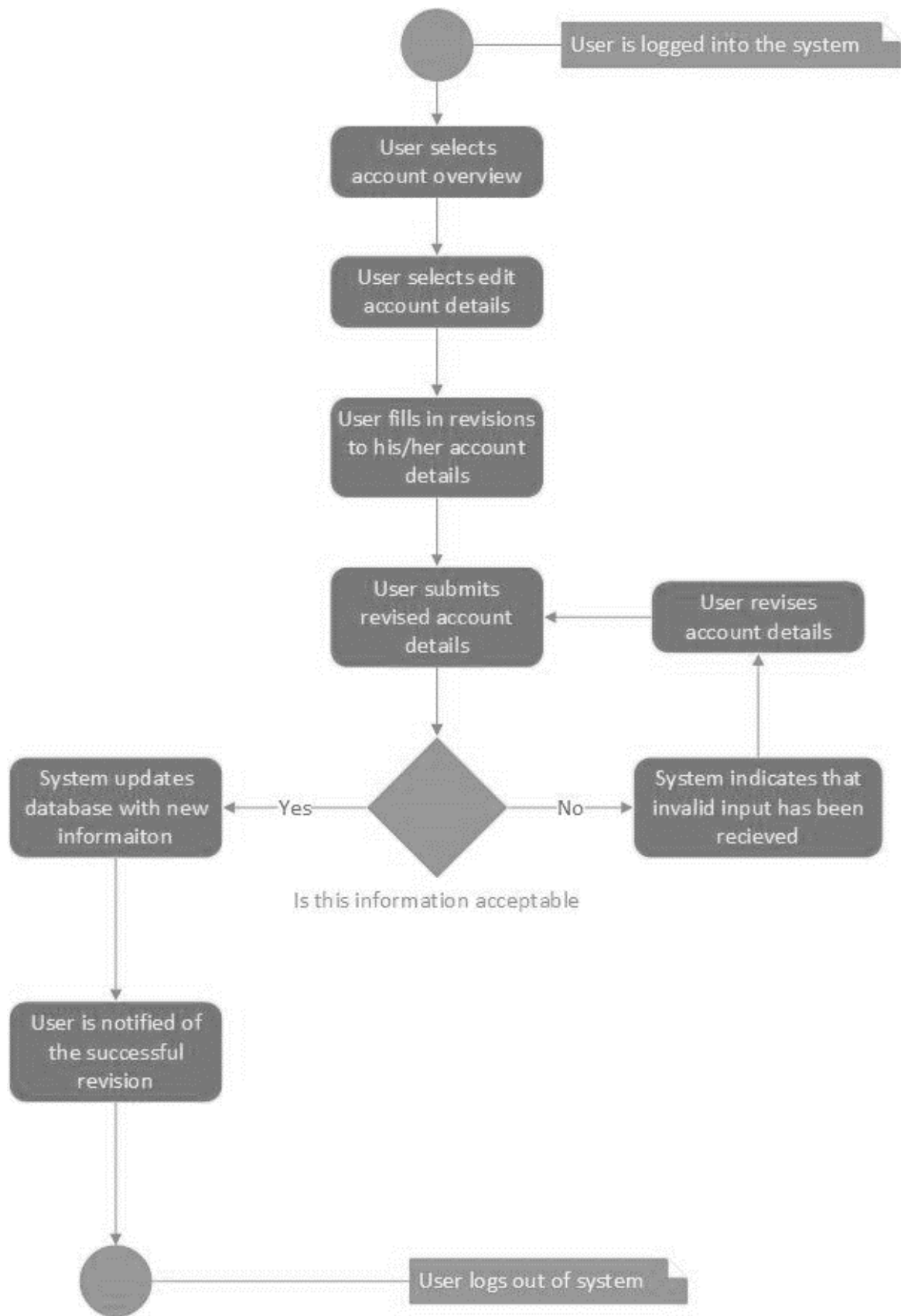
Request Time-off



File a Complaint



Edit User Details



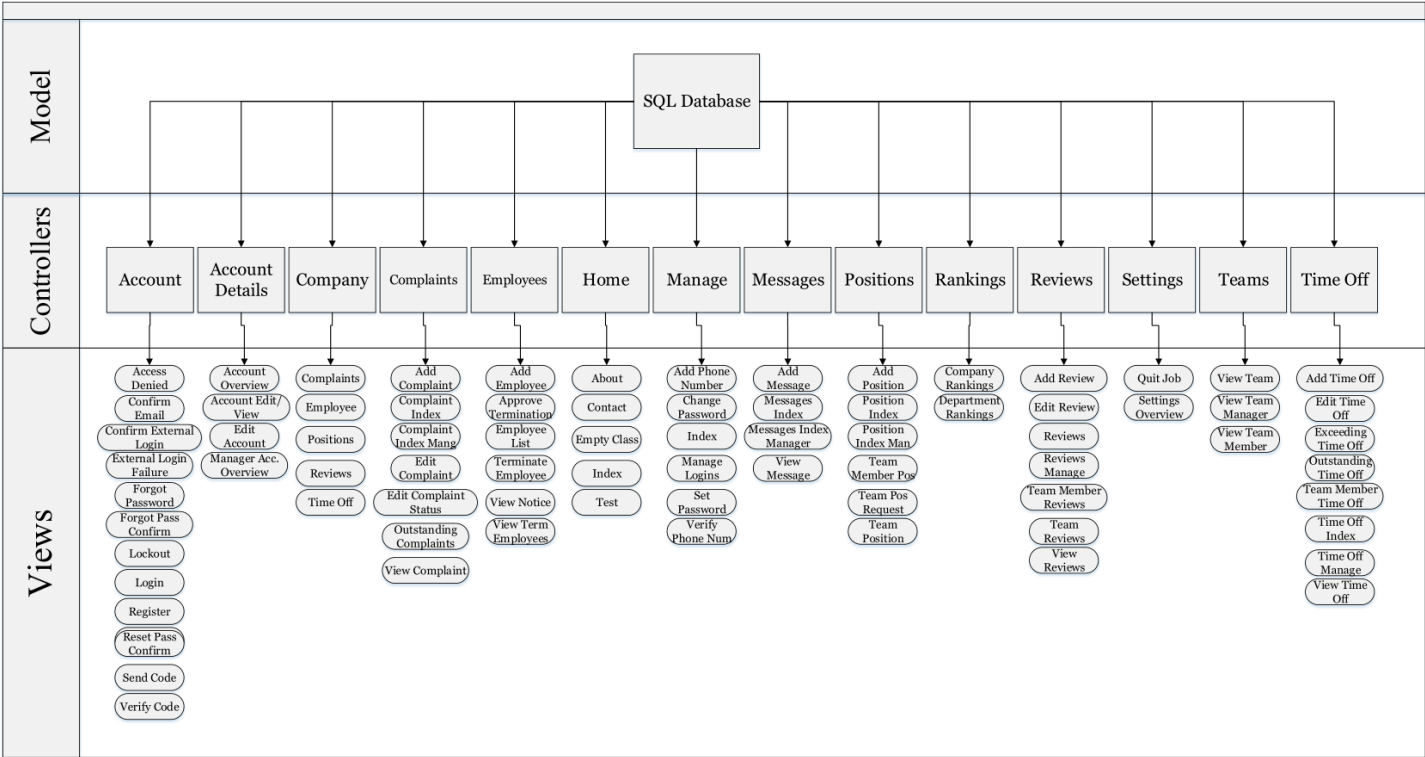
3. Design Overview

3.1 Introduction

For the system design, we opted to use a simple, well-tested framework, Microsoft’s MVC. The overall system architecture is composed of three separate tiers of code: Models, Views, and Controllers. Our system is designed to be as simple as possible and easily expandable in the future.

The Function Decomposition will serve as a general reference for the system’s current architecture. The initial design is subject to change, provide that not all use-cases can be met with the current design. If changes need to be made please contact the team and submit your request to the documentation engineer. Otherwise, ensure the application’s structure is exactly as the functional decomposition denotes.

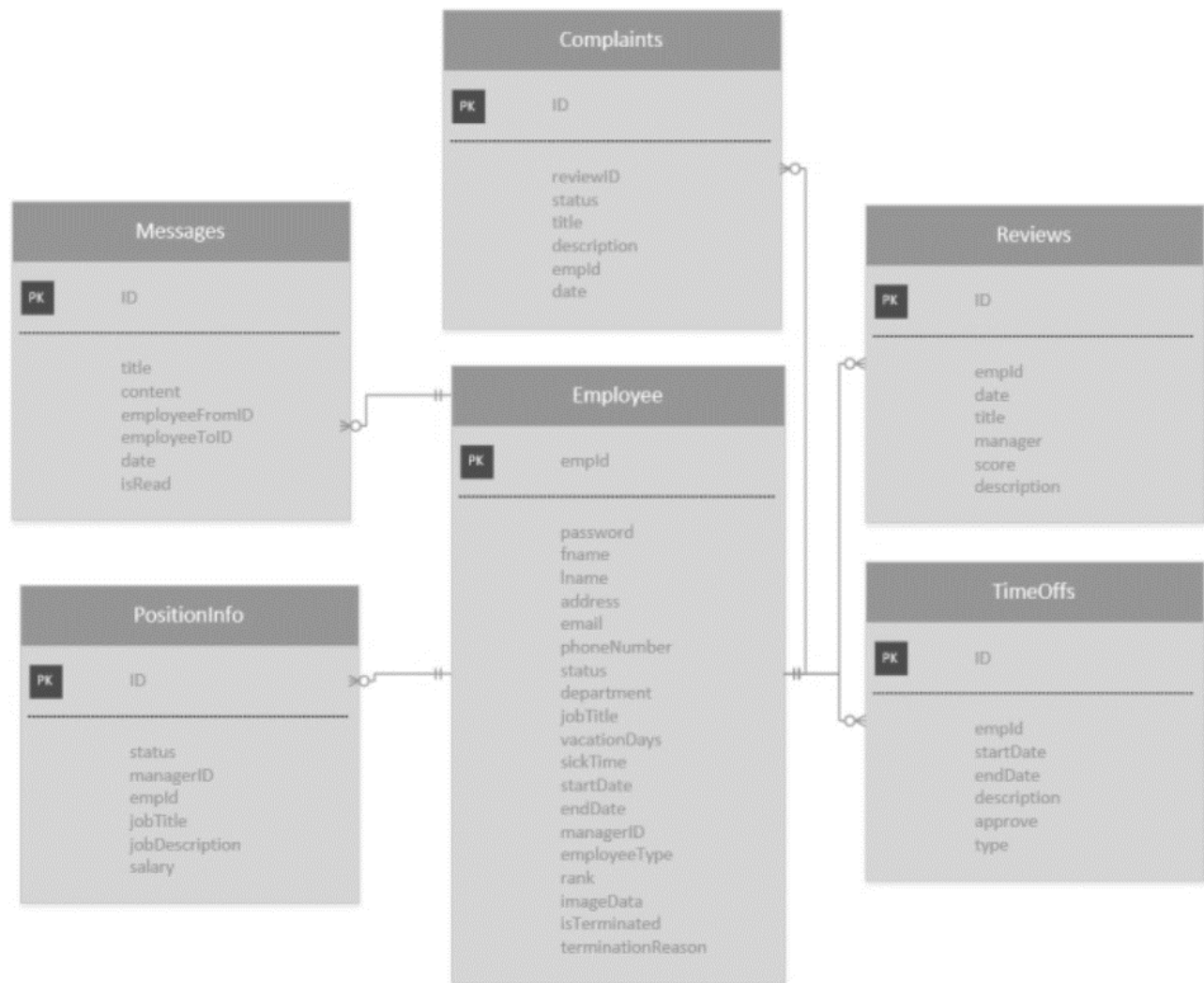
3.2 Functional Decomposition



4. Database Design

4.1 ER Diagram

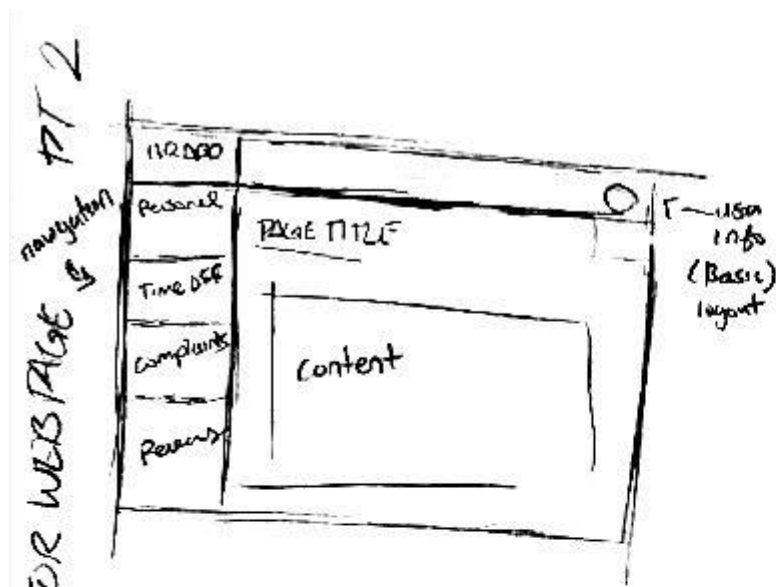
The following diagram represents all the tables needed in the SQL Database and their relationship to one another. If any design flaws are found or if you have a suggestion for improvements upon the design do not hesitate to notify the team members to discuss the potential improvement. Otherwise, be sure to include all of the following tables in the SQL Database as they appear in the ER Diagram below.



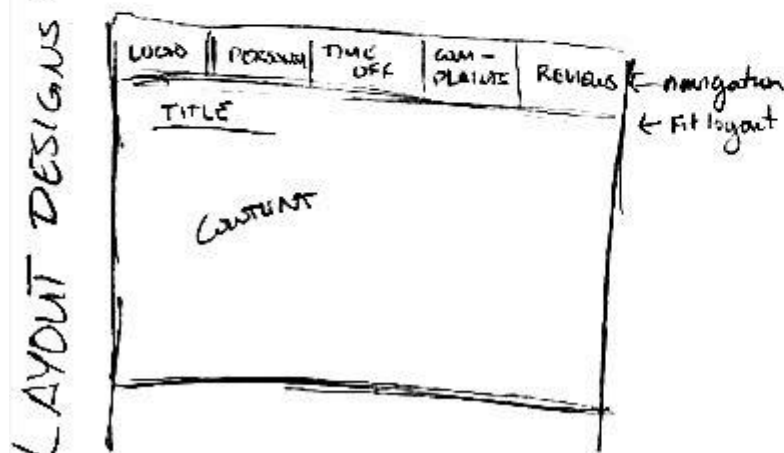
5. Wireframes

The main purpose of the HR Employee Tracking Application is to reduce the amount of man hours spent in employee management, allowing more time for employees to focus on their work. Therefore, we have decided to put a heavy focus on the user interface and user interaction. If the user is unable to find what he/she is looking for, or takes several clicks/attempts to complete their task, that is wasted time. Various user centered wireframes are displayed below; these should serve as a reference as to what the final product needs to look like and how it should behave.

5.1 Wireframe Sketches

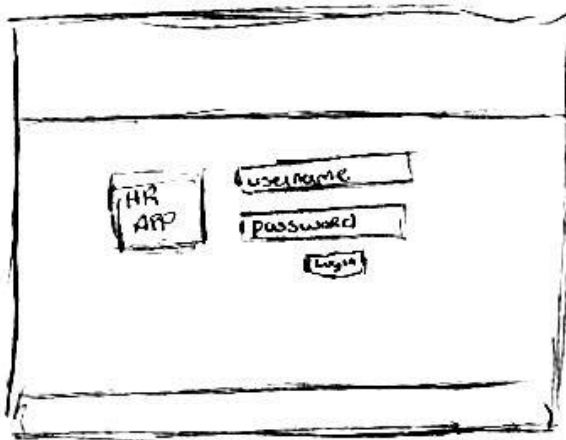


- Landing Page 3
- only the content changes from page to page.
 - Maybe collapsible nav?

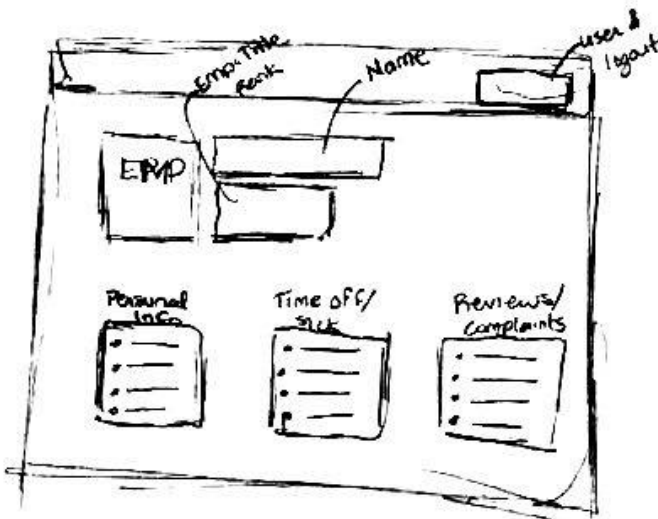


- Landing Page 4
- Content not static
 - Nav remains the same

DESIGN DOCUMENT FOR WEB PAGE PART 1

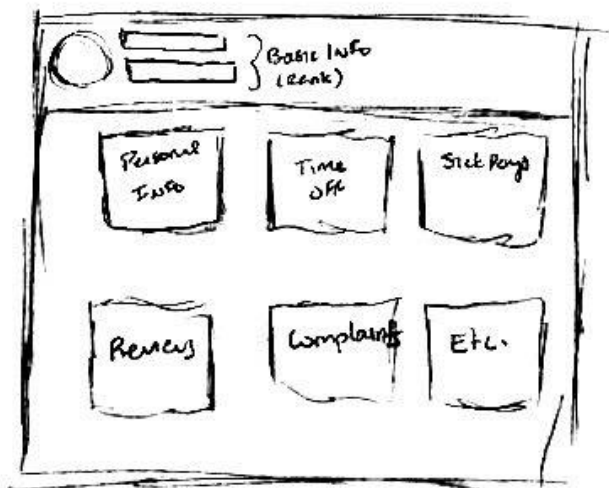


Homepage of Application



Landing Page 1

- All pages are listed on the landing page
- Categorized into basic categories for role
- More info in the top for users on landing page



Landing Page 2

- Pages have boxes that lead to more pages (categorizes the different kind of actions)
- Less info on pages

5.2 Finalized Wireframes

