Assignment #1 – Project Demo - PACT

Course code: 24F\_CST8319\_450

Facilitator: Moe Osman

Group Assignment

Project Proposal: PACT

***Group 11***

|  |  |
| --- | --- |
| **Name** | **Student number** |
|  |  |
| Abou-El-Kheir, Adam | 041139787 |
| Concepcion Perez, Michelle | 041044464 |
| Dinesh Ranaweera | 041090901 |
| Laperrière, François | 041125096 |
| Qingyi Zhang | 040545120 |

Date: October 8th, 2024.

# Assignment #1 - Planning & Calendar Tool (PACT)

# Introduction

The Planning & Calendar Tool (PACT) is a web-based application designed to simplify and streamline the process of scheduling appointments. This document outlines the project’s objectives, user deliverables, functionalities, technologies/frameworks, dependencies, and project timeline.

# 1. Project Objectives

PACT (Planning A Calendar Tool) is a web application designed to streamline the scheduling process. It allows clients to create, manage, and book appointments effortlessly, reducing the need for back-and-forth emails and improving time management for both personal and business use.

**Key Features:**

* **Calendar Sync:** PACT connects personal and professional calendars, offering a unified view of availability.
* **Custom Booking Links:** Clients can generate personalized booking links, enabling others to schedule meetings based on real-time availability.
* **Automatic Scheduling:** Meetings booked through PACT are automatically added to the calendar, removing the need for manual updates.
* **Easy-to-Use Interface:** The platform’s straightforward design ensures a seamless scheduling experience, minimizing the need for emails or phone calls.

**Benefits:**

* **Time Savings:** PACT reduces the time spent on scheduling, allowing clients to focus on more important tasks.
* **Increased Efficiency:** It streamlines the appointment process, cutting down administrative overhead.
* **Improved Communication:** The tool simplifies scheduling with clients, colleagues, and teams, enhancing overall communication.
* **Reduced Email Overload:** By automating the scheduling process, PACT minimizes unnecessary email exchanges.
* **Versatile for All Users:** PACT is suitable for individuals, teams, and businesses of any size.

# 2. User Deliverables

End User Deliverables:

Upon project completion, the following features will be delivered to the end user:

1. **Secure User Registration and Login**:  
   Users will be able to create accounts using their email and password. This will help keep their data private and make sure only they can access their scheduling information. The login process will be simple and include features like password recovery.
2. **Availability Management**:  
   Users will manage their availability with a simple calendar, selecting free days and times. They can block specific times for vacations or commitments to ensure accuracy and update availability as needed.
3. **Appointment Booking**:  
   Users can share a link with others to let them book appointments directly through the app. Once a time is chosen, the app will send both parties a confirmation and automatically update the user’s calendar. This will make scheduling much easier because there won’t be any need for back-and-forth communication. The app will also adjust for time zone differences.
4. **Personal User Dashboard**:  
   Each user will have a dashboard that shows all their upcoming appointments. The dashboard will be simple, allowing users to easily view, reschedule, or cancel appointments if needed. The dashboard will also show important notifications, like reminders for upcoming meetings or new bookings.
5. **Clean and Intuitive User Interface**:  
   The app will be designed to be simple and easy to use, so that anyone can quickly find what they need. Key features like availability management and appointment booking will be easy to access.

Complete List of Features:

* User Registration and Login: Secure account creation with email and password protection.
* Availability Settings: Manage available days and times through an interactive calendar.
* Appointment Booking: Share a link for others to book available time slots.
* User Dashboard: View, reschedule, or cancel appointments.
* Simple and Responsive User Interface: Easy navigation and clean design for efficient user experience.

# 3. Functionalities

Key Functionalities:

* User Registration and Login: Secure account creation and access management.
  + Registration:
    - Account Creation
    - Email Verification
    - Profile Setup
  + Login:
    - Authentication
    - Password Recovery
  + Session Management
    - Handling user sessions to maintain login state.
    - Option to logout and terminate sessions.
* Availability Settings: Users can specify their available days and times for meetings.
  + Time Slot Suggestions:
    - Display available time slots of other participants and meeting resources
  + Time Slot Selection:
    - Select from available time slots.
    - Move available time slots.
    - Delete available time slots.
* Appointment Booking: Users can view and book available time slots directly through the app.
  + Create new booking:
    - Select booking type.
    - Customize duration.
  + Notifications and Reminders
    - Send automatic confirmation to all participants.
    - Automated reminders sent via email.
  + Rescheduling and Cancellation
    - Allow users to reschedule their appointments.
    - Allow users to cancle appointments.
* User Dashboard: A personal dashboard displaying upcoming appointments and allowing easy management of availability.
  + List of upcoming appointments with summaries and due dates.
  + List of past appointments with summaries and due dates.
* User Friendly Interface: Intuitive and easy-to-navigate design for a seamless user experience.
  + Provide context sensitive online help.
  + Provide quick suggestions or brief descriptions when user hover mouse cursor over a button or highlighted terms.

# 4. Technologies/Frameworks

The PACT (Planning & Calendar Tool) project will use a mix of modern web technologies to build a secure, fast, and scalable application. These technologies follow industry standards and best practices, supporting both backend and frontend development.

**Backend Technologies:**

* **Java:** Java will be used to build the core features and manage server-side tasks. It is flexible and scalable, making it a solid choice for web application development.
* **Java Servlets**: Servlets will handle HTTP requests and responses, user sessions, and client-server communication.
* **JSP (Java Server Pages)**: JSP is will dynamically generate HTML content, ensuring seamless interaction between the frontend and backend.
* **Apache Tomcat**: Tomcat will be the server used to deploy and run the application. It works well with Servlets and JSP, providing a stable development environment.
* **Spring Boot** (Optional): If needed, Spring Boot can be used to simplify backend development by managing dependencies, creating APIs, and offering built-in tools for running and testing the application.

**Frontend Technologies**:

* **HTML & CSS**: These will be used to build and style the web pages, ensuring the design is responsive and functional on various devices.

**Database Technologies**:

* **MySQL**: MySQL will serve as the main database to store user data, schedules, and appointment details. It's a reliability in managing relational data makes it ideal for the PACT application.
* **JDBC:** JDBC will connect the Java application to the MySQL database. providing methods for querying and updating data.
* **Hibernate / JPA** (Optional): Hibernate or JPA can be used as an ORM tool to work with Java objects instead of SQL queries, streaming database management.

**Design Patterns**:

* 1. **Singleton**: Ensure that only one instance of a class, like the database connection manager, is created to save resources and allow global access to that instance.
  2. **Builder**: Assists in the step-by-step creation of complex objects, like user profiles, by separating object construction from its representation.
  3. **Model-View-Controller (MVC)**: Separates the application's logic, interface, and control. The model manages data, the view displays it, and the controller handles user actions.
  4. **Adapter**: Makes it easier to connect to third-party APIs without changing existing code.
  5. **Proxy**: Adds a layer of security by controlling access to sensitive data and services.

This mix of technologies will ensure that the PACT application is functional, secure, scalable and easy to maintain.

# 5. Dependencies

The PACT project relies on external libraries, frameworks, and services to function properly. These dependencies enhance the application’s features, simplify development, and improve deployment. Below is an overview of the main dependencies:

1. **JDBC Library**:

The JDBC (Java Database Connectivity) library connects the Java application to the MySQL database. It allows for efficient execution SQL queries, retrieval results, and database updates, essential for performing all CRUD (Create, Read, Update, Delete) operations.

1. **Spring Framework** (Optional):

The Spring Framework can be used for dependency injection and creating APIs. This will make backend development easier by reducing repetitive code and ensure a more modular structure.

1. **Apache Tomcat**:

Apache Tomcat is the servlet container required for deploying the web application. It handles the HTTP requests and responses, facilitating communication between the client and the server, while providing a reliable platform for Java Servlets and JSP.

1. **JWT (JSON Web Token) Library**:

The JWT library manages user authentication and session management. It securely sends information between users as a JSON object format, protecting sessions from tampering and unauthorized access.

1. **Hibernate / JPA** (Optional):

Hibernate or JPA will connect Java objects to database tables, allowing the application to manage data in an object-oriented manner. This improves efficiently and reduces repetitive code when handling complex relationships between data.

1. **Apache Maven** (Optional):

Maven is a tool that automates the building of projects and managing dependencies. By using a pom.xml file, it ensures all required libraries, are up to date and compatible, making the development process smoother.

1. **Email Service API** (Optional):

An external email service API, such as SendGrid or JavaMail, can be integrated to send automated appointment confirmations and reminders. This feature enables timely communication with users without needing an internal email server.

These dependencies are critical for adding specific functionalities to the PACT application. By incorporating these tools and services, development becomes smoother, and the application gains enhanced features like secure authentication, efficient database handling, and automated email notifications.

# 6. Project Timeline

* Week 1-2: Requirement analysis and project planning.
* Week 3-4: Backend development using Java, Servlets, and Apache Tomcat.
* Week 5-6: Frontend development with HTML and CSS.
* Week 7-8: Database integration using JDBC, MySQL, Hibernate or JPA.
* Week 9-10: Implementation of security features (HTTPS, JWT, Auth0).
* Week 9-12: Testing and debugging of the application.
* Week 13: Final review and deployment.

# Conclusion

PACT aims to provide an efficient, secure, and user-friendly platform for scheduling and managing appointments. By leveraging modern web technologies and design patterns, the project seeks to enhance productivity and simplify time management for its users.