
Eclectic Computing

**Time Tracking Tool
Software Architecture Document**

Version <1.3>

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

Revision History

Date	Version	Description	Author
7/Mar/05	1.0	Documentation of high-level architecture.	David Janzen
7/Apr/05	1.1	Architecture at end of Elaboration	David Janzen
11/Apr/05	1.2	Completed Architecture at end of Elaboration	David Janzen
6/May/05	1.3	Updated diagrams to reflect implementation decisions	David Janzen

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

Table of Contents

1.	Introduction	4
1.1	Purpose	4
1.2	Scope	4
1.3	Definitions, Acronyms, and Abbreviations	4
1.4	References	4
1.5	Overview	4
2.	Architectural Representation	4
3.	Architectural Goals and Constraints	4
4.	Use-Case View	4
4.1	Use-Case Realizations	4
5.	Logical View	5
5.1	Overview	5
5.2	Architecturally Significant Design Packages	6
6.	Interface Description	9
7.	Size and Performance	10
8.	Quality	10

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

Software Architecture Document

1. Introduction

1.1 Purpose

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions that have been made on the system. This document is intended to guide the construction phase of the project.

1.2 Scope

This Software Architecture Document provides an architectural overview of the Time Tracking Tool. This tool is a generic, web-based tool that enables users to enter time information on tasks completed as part of projects with tasks and sub-tasks. The tool includes an interactive timer for automatically collecting time information. Information will be stored in a database. Administrative and managerial functions will be provided including configuring projects/tasks/subtasks, and generating reports.

1.3 Definitions, Acronyms, and Abbreviations

Refer to Project Glossary.

1.4 References

Refer to Use Case Specification and the Analysis Classes documents.

1.5 Overview

This document serves as the primary documentation of the software architecture. However, not all views will be documented here to avoid redundancy and ease document maintenance. In particular, the Use Case View will be documented in the Use Case Specification. Architectural Representation

The architecture will be documented with a Use Case View, Logical View and an Implementation View. The Use Case View will be documented in the Use Case Specification document, but a sampling of use-case realizations will be documented in this document.

2. Architectural Goals and Constraints

The Time Tracking Tool is a web application that will use a single Page Controller for all interaction between the html user interface and the server. The user interface will also use a Java applet for a timer feature and local storage on the client machine. This applet will interact with a separate controller servlet on the server.

This application is not considered life critical and is not needed to support an inordinately large number of users. Simple user name and password permissions will provide minimal security. Encryption is not considered an essential requirement.

3. Use-Case View

This section documents a few typical use case realizations.

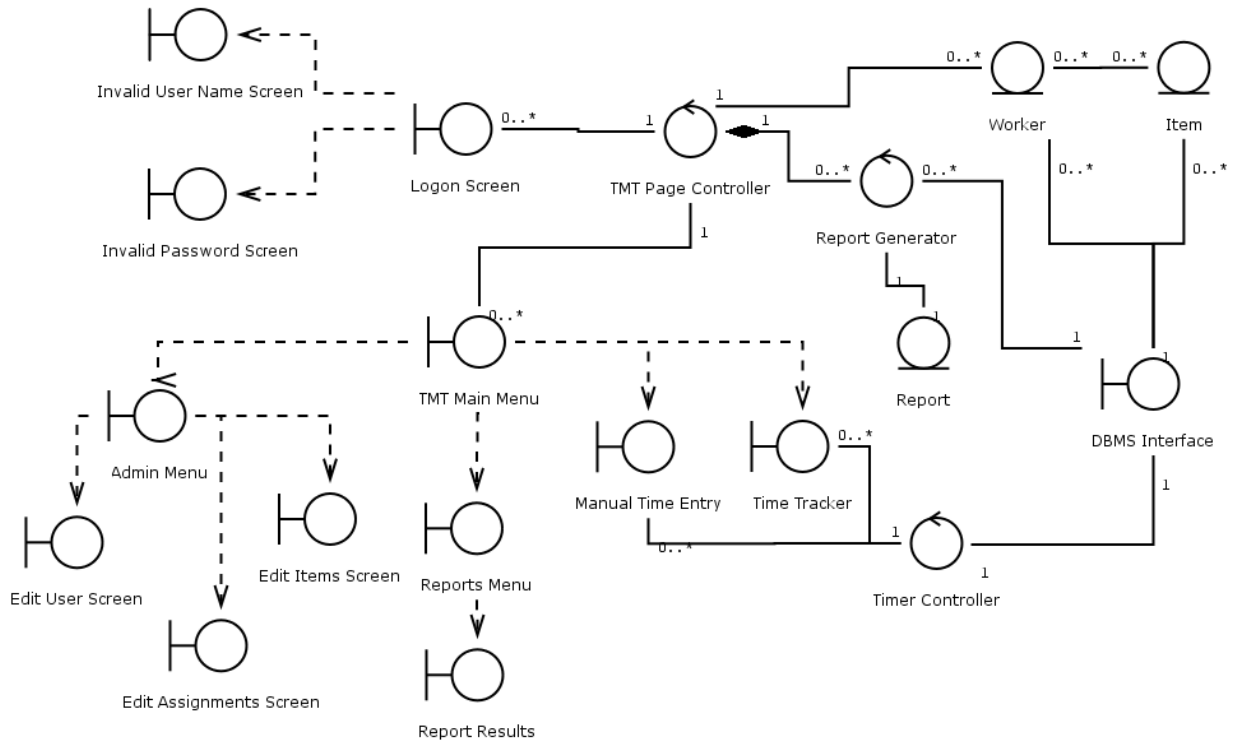
3.1 Use-Case Realizations

See Time Tracking Tool Use Case Realization Specification document.

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

4. Analysis Classes

The following diagram contains the analysis classes for this project.



5. Logical View

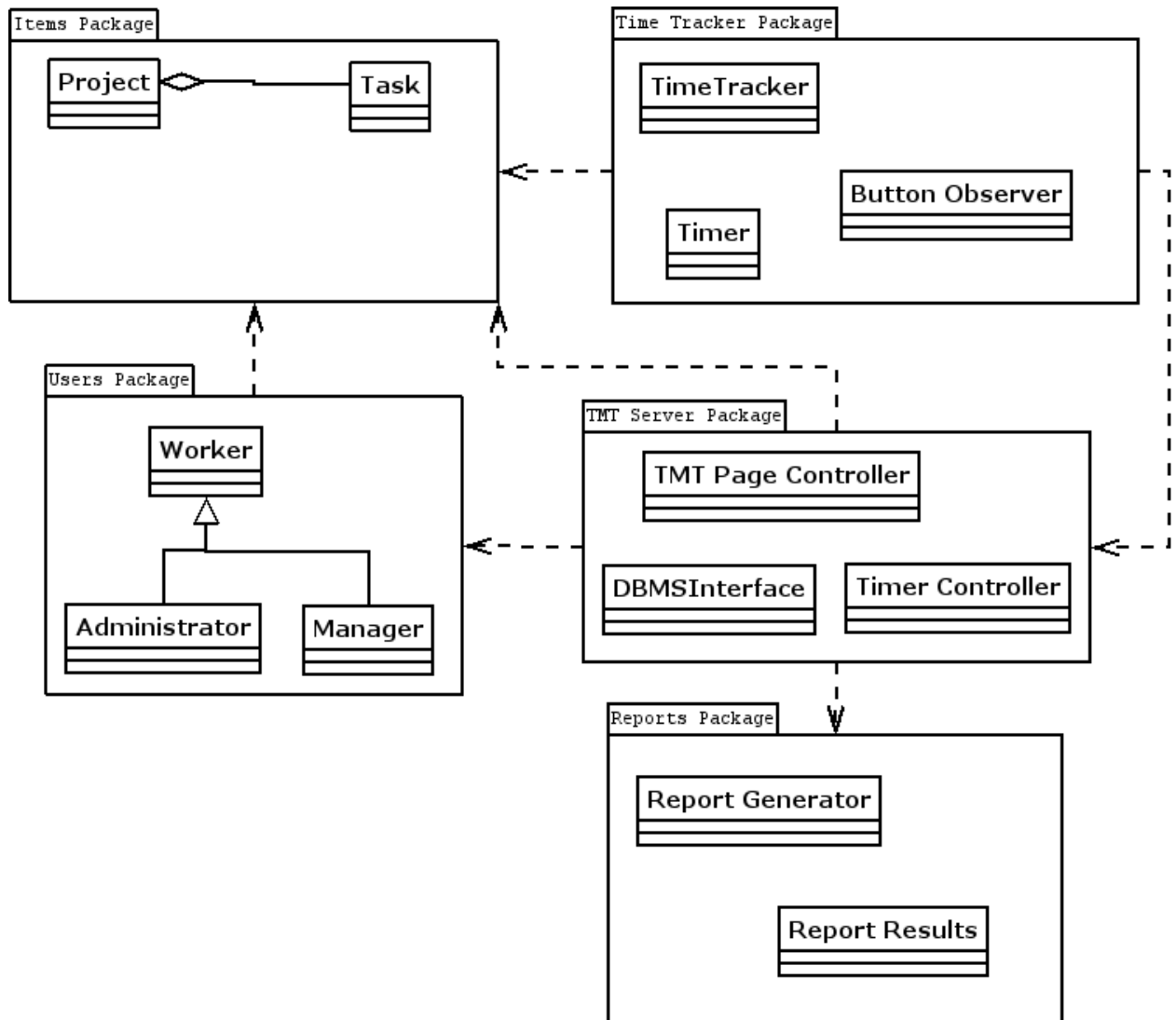
This section describes the architecturally significant parts of the design model. The architecturally significant design packages will be described, followed by a more detailed description of the design model classes.

5.1 Overview

This subsection describes the overall decomposition of the design model in terms of its package hierarchy and layers.

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

5.2 Architecturally Significant Design Packages



6. Implementation View

This section describes the architecturally significant parts of the design model including its decomposition into subsystems and packages.

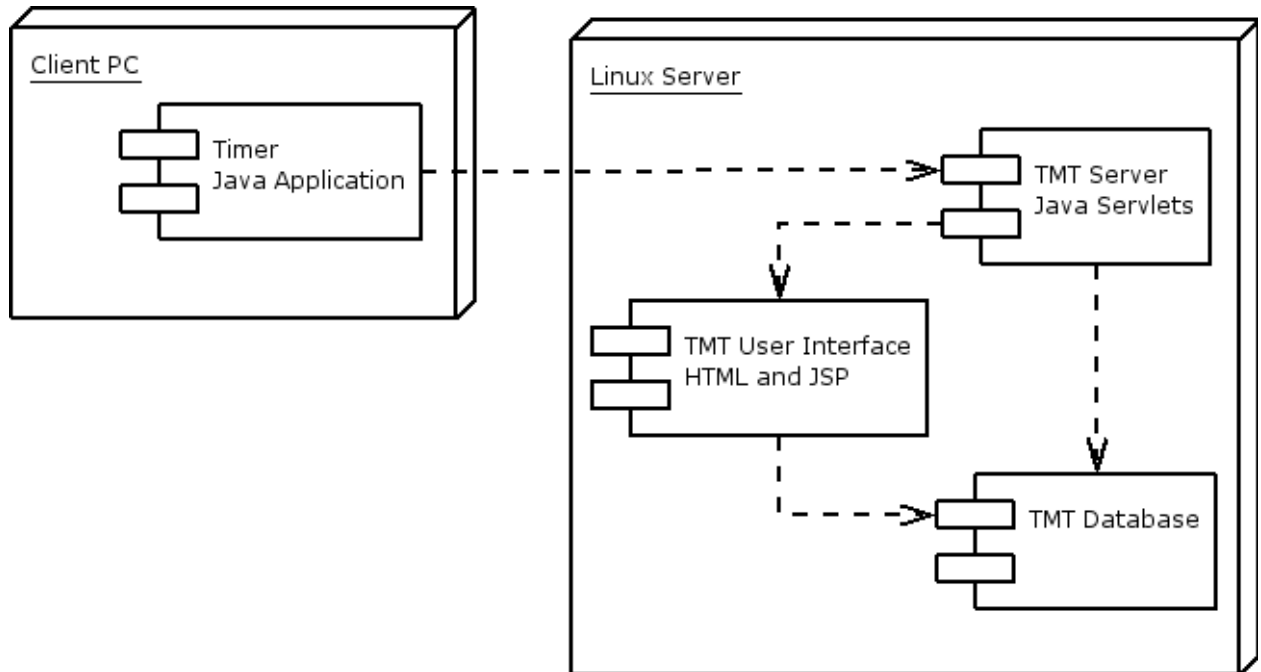
6.1 Overview

This system will consist of client, server, server-side user-interface, and database components. The client will utilize a standard web browser. The server-side user-interface component will use JavaServer Pages to generate html user-interface files to be sent to the client. One Java GUI application will also be available through the Java WebStart technology. This assumes a Java run-time environment will be installed on all client machines. Http will be used for communication between the client and server components. The server component will be built with Java Servlets and will use JDBC to communicate to a MySQL database.

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

6.2 Architecturally Significant Components

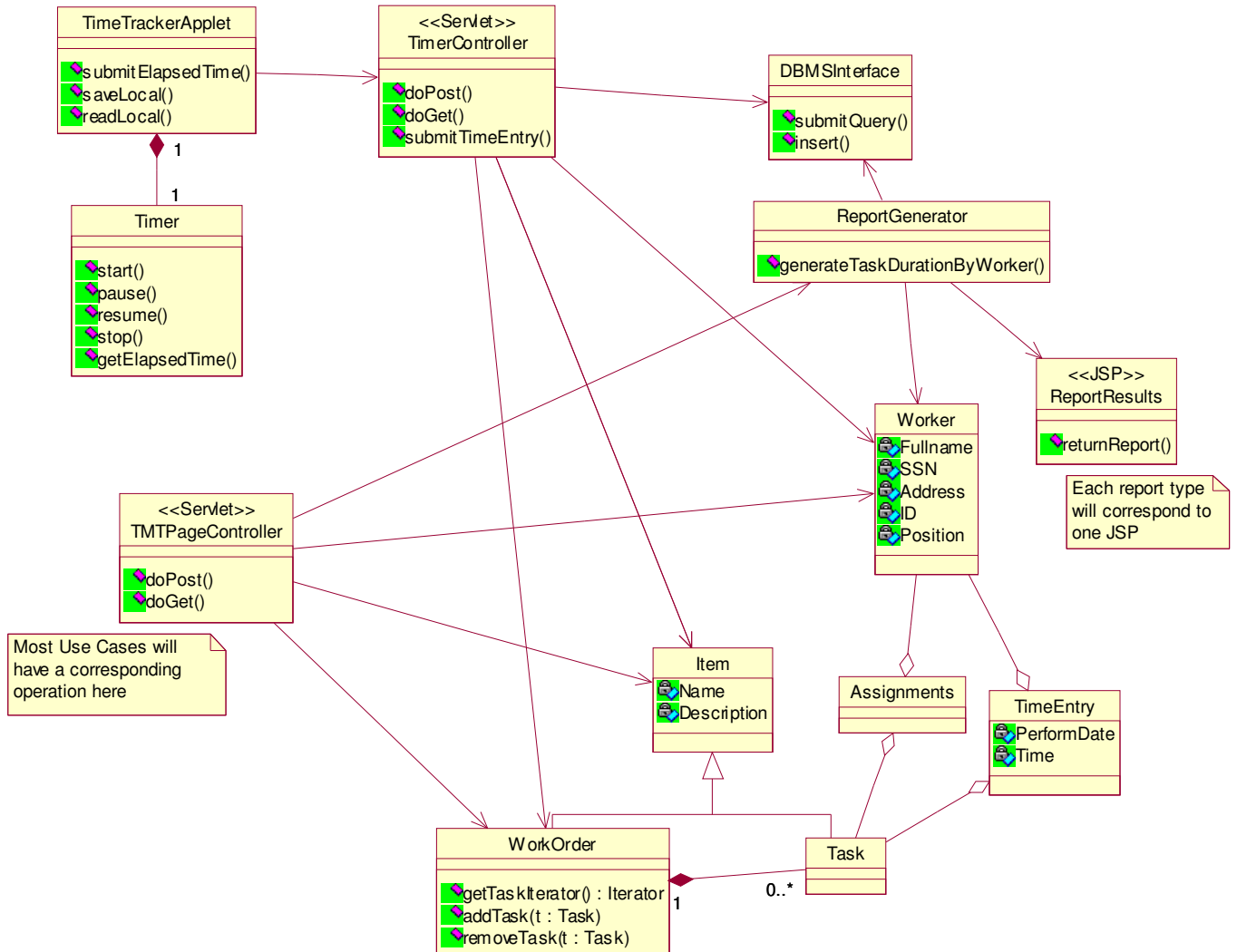
The following constitutes the Deployment View.



Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

6.3 Architecturally Significant Design Classes

The following are a subset of architecturally significant design classes.

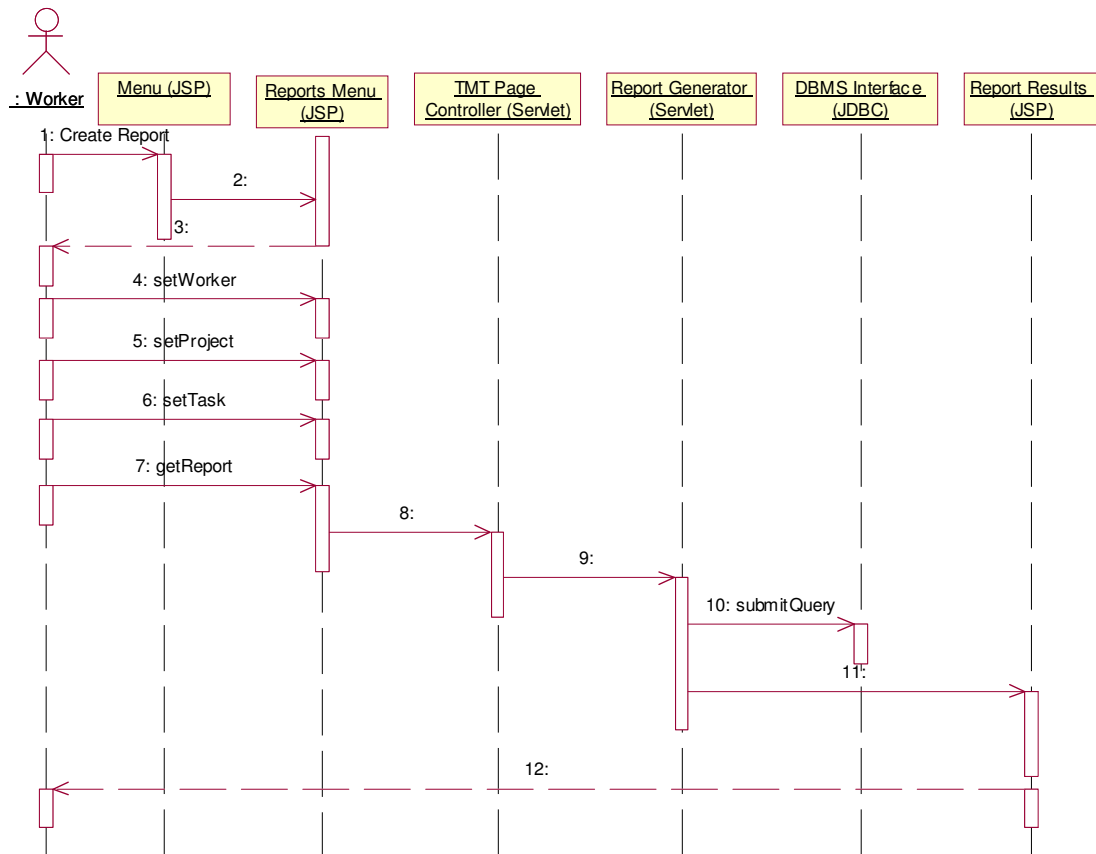


Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

6.4 Architecturally Significant Interactions

The following is one example of a architecturally significant interactions between classes. Additional interaction diagrams are located in the Use Case Realization document.

Create Task Duration for Worker Report:



7. Interface Description

The user interface will be a combination of a web-based interface and a stand-alone graphical user interface. User Interface screen mock-ups are provided in the following files:

- Screen 1 – Login
- Screen 2 – Menu
- Screen 3 – Time Tracker
- Screen 4 – Reports
- Screen 5 – User Admin
- TTT UI Report Prototype

Time Tracking Tool	Version: <1.3>
Software Architecture Document	Date: <06/May/05>
SwArch.doc	

8. Size and Performance

This application is not needed to support life-critical applications or a large number of users. Similarly, load is not expected to be substantial. As a result, no particular architectural aspects have been added to support load or throughput. Apache Tomcat and MySQL will be used as the Java Servlet engine and database management system. Both are expected to provide adequate performance.

9. Quality

The choice of using a Page Controller assumes a relatively small number of Use Cases. While this approach has several advantages for small applications, it could become constraining if the number of use cases increases substantially. Other controller-based architectures are available for larger applications. The choice of using a web browser based client with Java WebStart will enhance application portability. Access to read and write local files does introduce a security concern on the client. This application will need to be marked as trusted so it is not expected that this system will be used publicly.