Town Planning

SRS Document

Disclaimer

This Software Requirements Specification document is a guideline. The document details all the high level requirements. The document should be used as a guideline by the students to design the Solution Architecture for the project. The document also describes the broad scope of the project and high level logical object model. But while developing the solution if the developer has a valid point to add more details being within the scope specified then it can be accommodated after consultation.

Town Planning System

Introduction

The purpose of this document is to define scope and requirements of a 'Town Planning' for application available to builders for request processing online that eliminates the need to reduce administrative effort that goes in submitting paper work and tracking of the requests.

This document should be used by the development team to architect the solution the project.

Management Summary

The building plan approval process relates to the issue of permission for the construction of buildings based on specific set of rules and regulations. As per the relevant provisions of the State Municipal Acts any person desirous of undertaking a construction activity (fresh construction as well as modifications to existing structures), is required to obtain prior sanction from the relevant local body before embarking on the same. In the local bodies, the town planning section is responsible for issuing the permissions to applicants. It also called building license in many states.

The main functions of the town planning section, as relating to building plan approval process, are both regulatory and developmental oriented. They are as follows:

1. On the regulatory front the functions include formulation of building rules, master plan rules and zonal regulations.
2. On the developmental side the functions include: Implementation of master plans, Road widening programs.
3. Issuance of permission for buildings and layouts.
4. Regularization and demolition of unauthorized construction.
5. Removal of encroachment

The proposed solution will be designed & developed to run on IBM WebSphere Application Server and IBM DB2 Universal Database in a 2-tier architecture.

Key Assumptions

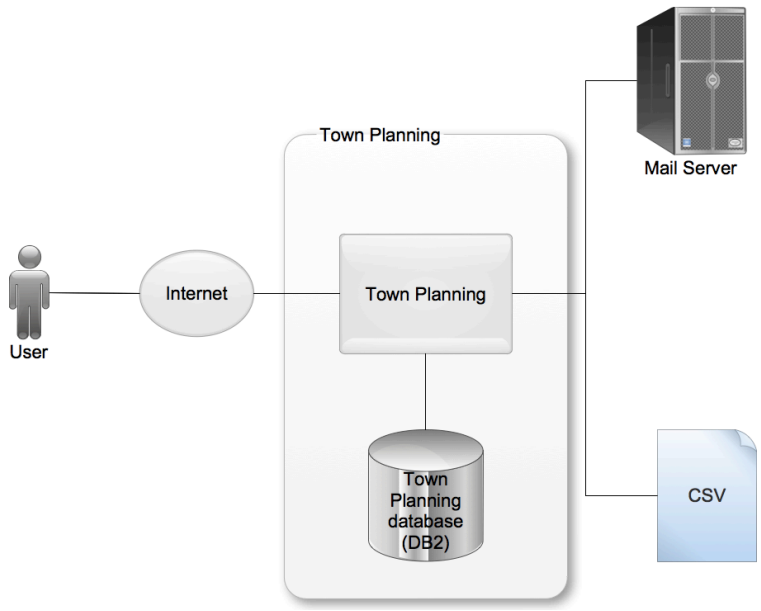
1. The developer of this project should be aware of challenges of municipal systems in a metropolitan environment.
2. Software developers must consider the existing Portal and Accounting systems, when designing this solution, since there must be seamless integration with these systems.

3. Builders will have a login for the system usage via registration method. To be handled outside the system.

High Level Architecture

Town Planning's high level architecture is illustrated through the context diagram shown below. It will have following categories of users:

1. Citizen
2. Town Planning Officer
3. Planning Assistant
4. Builder



Town Planning Context Diagram

Town Planning	The main functions of the town planning section, as relating to building plan approval process, are both regulatory and developmental oriented. The building plan approval process relates to the issue of permission for the construction of buildings based on specific set of rules and regulations.
Town Planning Database	Requests and processing data are stored along with record tax policies etc.
CSV	Masters such as Locations, Inspection Officer, Builders are uploaded via CSV format.
Mail Server	All notifications are routed through the Mail Server

Functional Requirements

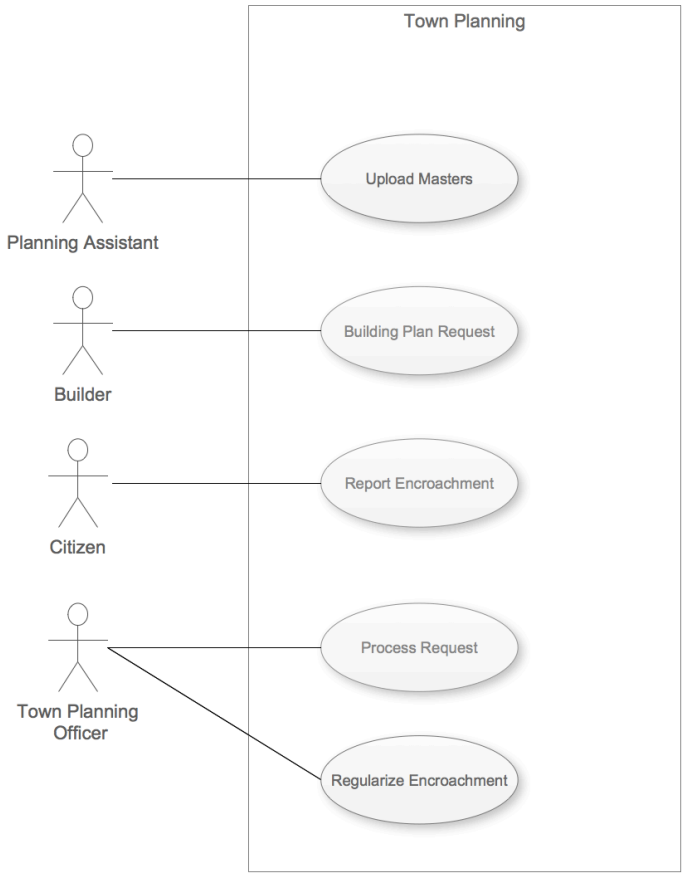
The high level functional requirements for the Town Planning are outlined in the Use Case diagram described in this section.

Town Planning will provide a secure user-id/password based secured login mechanism to access its services. The details of this are not outlined here. The development team is expected to create these keeping in mind the general practices followed by the web applications. Login will be a prerequisite to use Town Planning. Internal users will be provided user id/password pair separately.

Once user logs in, menu options shall primarily come from the use case of the role player.

Use Case Diagrams

The following figure illustrates the Use Case diagram for the system. The MIS use cases are not detailed here.



Use Case Diagram

Use Cases

Upload Masters

Use Case Element	Description
Number	UC.01

Use Case Element	Description
Application	Masters in Town Planning are uploaded using CSV Format Location master will have location id, location name Builders master will have builder id, builder organization name, contact person, mobile, email id. Inspector Master will have Inspector id, Inspector Name, Location id, Phone Number, Email id.
Use Case Name	Upload Masters
Primary Actor	Planning Assistant
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Upload Masters** menu item on the landing page
Basic Flow	System prompts for the file name to be uploaded. Standard file upload dialog is presented to select a file from the local system. • The selected file data is uploaded in the related tables; if an existing record is encountered, the old details are replaced with the new details.
Alternate Flow	• In event of incorrect CSV format, system gives an error and NO data is uploaded. • Operation is cancelled
Output	System displays the number of records uploaded. It also highlights the number of records updated (i.e. already existing ones being replaced)

Request for Building Plan

Use Case Element	Description
Number	UC.02
Application	The building plan approval process relates to the issue of permission for the construction of buildings based on specific set of rules and regulations.
Use Case Name	Request for a Building Plan
Primary Actor	Builder
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Request for Building Plan** link on the landing page

Use Case Element	Description
Basic Flow	• The system displays a form to capture the following information. All fields are mandatory. • Project Name, Duration in Months, Total Estimated Value. • Attach Documents : Registered title deed duly attested by a gazette officer. Soft Copy of Site plan prepared by licensed surveyor. • Following Attachment options to be checked by the user for validating the inclusion of critical information for processing. a) Boundaries of land as per certified survey records with survey number indicated, b) Alignment of proposed streets, c)Proposed sizes and number of plots, d)Places set apart for public purposes (as mentioned in the relevant act), e)Electric and telegraph lines, f) water mains, g)sewers, h)alignment with major highways, i)roads passing through the layout. Click on submit to send the request for processing. Details entered and uploaded are saved with request id for the logged in builder. Request moves to 'For processing' State.
Alternate Flow	Cancel will abandon the operation with no impact on database.
Output	None

Process Request

Use Case Element	Description
Number	UC.03
Application	Process Building Plan Request for further inspection
Use Case Name	Process Building Plan Request
Primary Actor	Town Planning Officer
Secondary Actor	None
Pre-condition	None
Trigger	The user clicks on the **Process Request** link on the landing page
Basic Flow	• The system displays the list of building plan requests in the 'processing queue'. User selects one request, the system displays details as entered by the builder. • The user selects the inspector form the pick list for allocating the inspection work. • TPO clicks on 'Accept' to begin the request processing activities. System updates the Building register. The application is allotted to selected building inspector. The system sends email notification to the applicant and building inspector with the acceptance number and site details. • The request moves to 'Inspection state'
Alternate Flow	User selects 'Reject' to discard the request for approval of plan for processing
Output	Email notifications in both cases of action

Report Encroachment

Use Case Element	Description
Number	UC.04
Application	The citizens have the right to complain about the encroachments by builders or individuals.
Use Case Name	Report Encroachment
Primary Actor	Citizen
Secondary Actor	None
Pre-condition	None
Trigger	The user clicks on the **Report Encroachment** link on the landing page.
Basic Flow	• The system displays the form for entering information. All fields are mandatory • Location pick list, Encroachment description, Affected Building, Appearance Date, Citizen contact details(address, mobile, email id) • Attach image of the encroachment. • The form is saved on 'Submit' the request id is generated and attached to the record. User is notified via email. The request moves to 'Pending for processing' state.
Alternate Flow	Pressing Cancel abandons operation, no database gets affected
Output	None

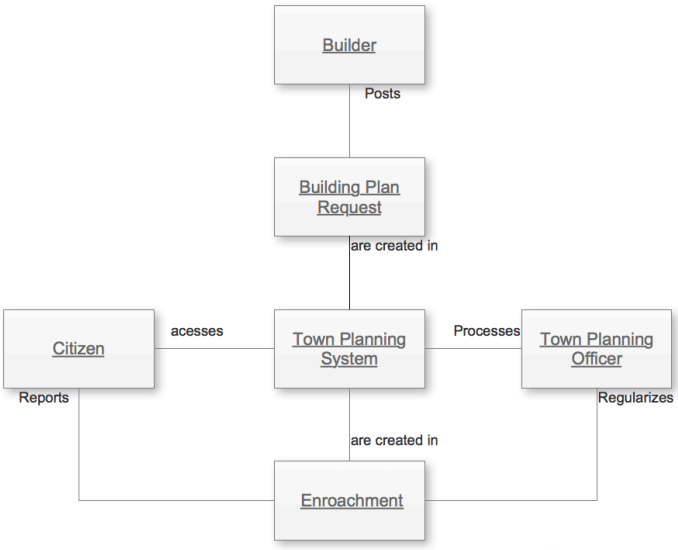
Regularize Encroachment

Use Case Element	Description
Number	UC.05
Application	The encroachments are unauthorized and require action by the Municipal bodies to ensure citizens do not face difficulties in their day to day activities.
Use Case Name	Regularize Unauthorized Construction
Primary Actor	Town Planning Officer
Secondary Actor	None
Pre-condition	None
Trigger	The user clicks on the **Regularize Encroachment** link on the landing page.
Basic Flow	• System displays a list of pending requests. User selects a single request to process. The details entered by the citizen are displayed. • TPO selects from the list of the next actions (Physical Inspection due / Invalid Request / Verify with Building Plans/ Demolition ordered/ Completed) • Action Date : Date picker • The request also moves the state as selected in the action. This use case gets activated number of times for the same request till status becomes completed.
Alternate Flow	Pressing Cancel abandons operation, no database gets affected
Output	email notification to citizen and inspectors

Note: Use cases to track status of requests are not given, in case time permits, the developer may consider developing feature for request tracking.

Logical Object Model

A high level logical object model of the system is shown below. During technical design it will be transformed into a physical model covering all system entities. Such a diagram will include their relationship and its cardinality.



Logical Object Model

1. Town planning system has Builders, Citizens as external users. Town planning officer is an internal user of the system
2. There are two kinds of requests handled in the system, a) Building Plan Request and b) Encroachment removal.
3. The Building plan request is raised by the registered Builders (uploaded as CSV) in the system.
4. Citizens of the country can provide their contact details and report about encroachments that are hindering public routine activities.
5. Town Planning Officer processes the requests in the system. Complete upward hierarchy of processing has not been included to contain the scope of project for students.

Database Design Guidelines

This involves the transformation of the use cases, state diagrams, and logical object model into detailed and optimize physical database table designs.

Typically persistent classes will map to table(s) with their attributes as columns of the table. In some cases a high level object may map in to a master-child table. Invoice is one such example where it maps in to "invoice_header" and "invoice_line_item" table.

Associations between two persistent objects are realized as foreign keys to the associated objects. A foreign key is a column in one table that contains the primary key value of the associated object.

Similarly, a standard technique in relational modeling is to use an intersection entity to represent many-to-many associations. Following is a broad checklist for physical database database design:

1. Database must be properly normalized except those instances where de-normalization help improves performance. This option must be used with special care.
2. All persistent classes that use the database for persistency must map to database structures.
3. Many-to-many relationships must have an intersecting table.
4. Primary keys should be defined for each table, unless there is a performance reason not to define a primary key.
5. Indexes should be defined to optimize access.
6. Data and referential integrity constraints should be defined.

Testing Approach

Quality of the software can be achieved with basic hygiene and consistency followed during design and development of User Interface(UI), Navigation, Validations as per the business process requirement.

To ensure the project delivers acceptable quality to the customer, its important to create a checklist of the conventions to be followed across. Common checks as below are for your reference during design and development:

Common Checks	Validation Type
Page Title is valid for the feature being provided on the page	UI
Order of the Data Entry Fields is logical as per the functionality being provided by the feature	UI
Order of the Display only Fields makes viewing and understanding easy for the user	UI
Spellings and Correctness of Label for the Data Entry and Display fields	UI
The labels are not wrapping onto another row thereby adding a blank row on the page	UI
The fields with drop down are displayed in single row instead of drop down coming on the next row	UI
Data Entry field basic validations are working i.e Text field /Numbers / Dates allow data for their type only	Functional
The dates are following a standard format dd/mmm/yyyy on all forms	UI
The color scheme of all forms i.e headers labels , alerts, entry fields are uniform throughout the application	UI
The action buttons for a New Data Entry Form are uniform for all forms that is allowing data entry	UI
The action buttons are performing the desired action e.g. "submit" is creating a new record if there are no errors and recording all the input fields, whereas 'cancel' is not creating a new record in the database	Functional
The links provided on the forms are opening correctly.	Functional
The data feed mechanism for Read and Write files is generating a log with count of entries.	Navigation

Suggested Technical Reading

The project is aimed at making the student understand concepts of Design and Development using IBM Rational tools, Web Sphere Application Server and DB2 Database. The following reading reference is easy to understand and should be read to get a clear understanding of capabilities of the tools and how you would leverage them to execute a project.

Technical Reference	URL to access
RAD - Tackling challenges of software development with Rational Application Developer for WebSphere Software	http://www.ibm.com/developerworks/rational/library/08/0926_ackerman-mahate/index.html
IBM Education Assistant - Rational Application Developer 7.5	http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/index.jsp?topic=/com.ibm.iea.rad_v7/rad/rad75.html
RSA-Overview of Rational Software Architect for WebSphere Software Version 7.5	http://www.ibm.com/developerworks/rational/library/08/0926_arnold/index.html
Using the new features of UML Modeler in IBM Rational Software Architect Version 7.5	http://www.ibm.com/developerworks/rational/library/08/0926_diu/index.html
Rational Technical Library	http://www.ibm.com/developerworks/rational/library/