University of Tartu
Faculty of Mathematics and Computer Science
Institute of Computer Science

Project
Scenario

MTAT.03.229 Enterprise System Integration

Authors:
    Ardi Aasmaa (A84120)
    Kadri Oluwagbemi (B04877)
    Kristjan Veskimäe (A20792)
    Polad Mahmudov (B79594)
    Tural Ismayilov (B79598)

Supervisor:
    Luciano García-Bañuelos

Tartu 2018

Table of Contents

Introduction

Plant hiring is the equipment rental from specialized rental companies to construction companies. In this project, we model and implement the procurement process via interconnected enterprise systems in rental and building company, called respectively RentIt and BuildIt.

In general, the design was kept as simple as possible. Due to time limitations, handling edge cases is out of scope from this project. Basic functionality, however, is provided for the requirements by this solution.

Part I: BuildIt

In this first part, we present the models for building company domain.

Domain model

First we present domain model in **Figure 1**. Identifiers are omitted from this conceptual model, except for PlantInventoryEntry and PurchaseOrder that have external URL as identifier. PIE rental price is assumed to vary and is therefore not persisted into the DB. For simplicity, PHR can only be extended once.
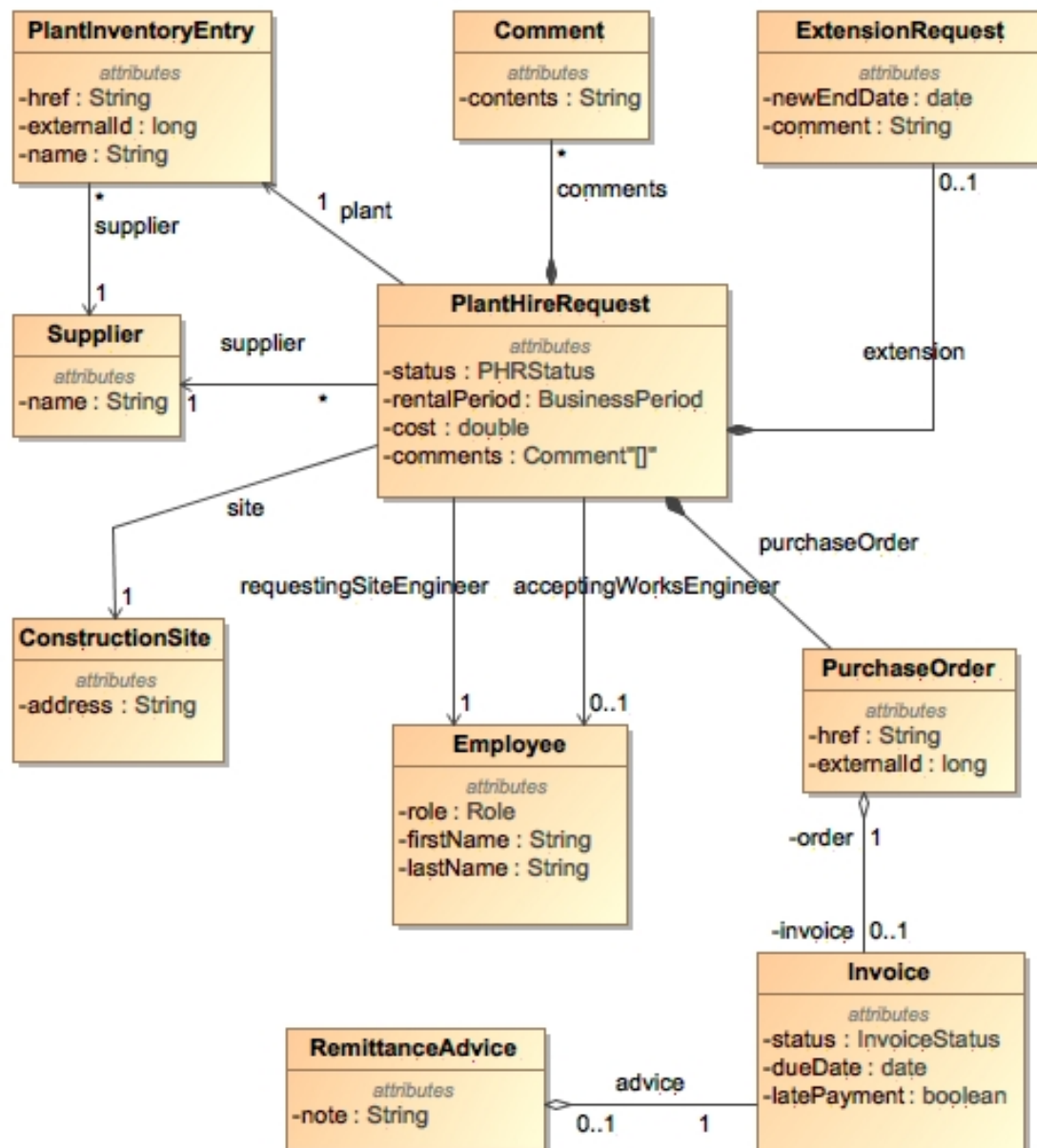

Figure 1 BuildIt domain model

The corresponding data types for BuildIt are in **Figure 2**.

package BuildIt[BuildItDataTypes]

«dataType»
BusinessPeriod

attributes
-startDate : date
-endDate : date

«enumeration»
Role

enumeration literals
SITE_ENGINEER
WORKS_ENGINEER

«enumeration»
InvoiceStatus

enumeration literals
PENDING
ACCEPTED
REJECTED

«enumeration»
PHRStatus

enumeration literals
PENDING_WORKS_ENGINEER_APPROVAL
PENDING_RENTAL_PARTNER_APPROVAL
ACCEPTED_BY_RENTAL_PARTNER
REJECTED
CANCELLED
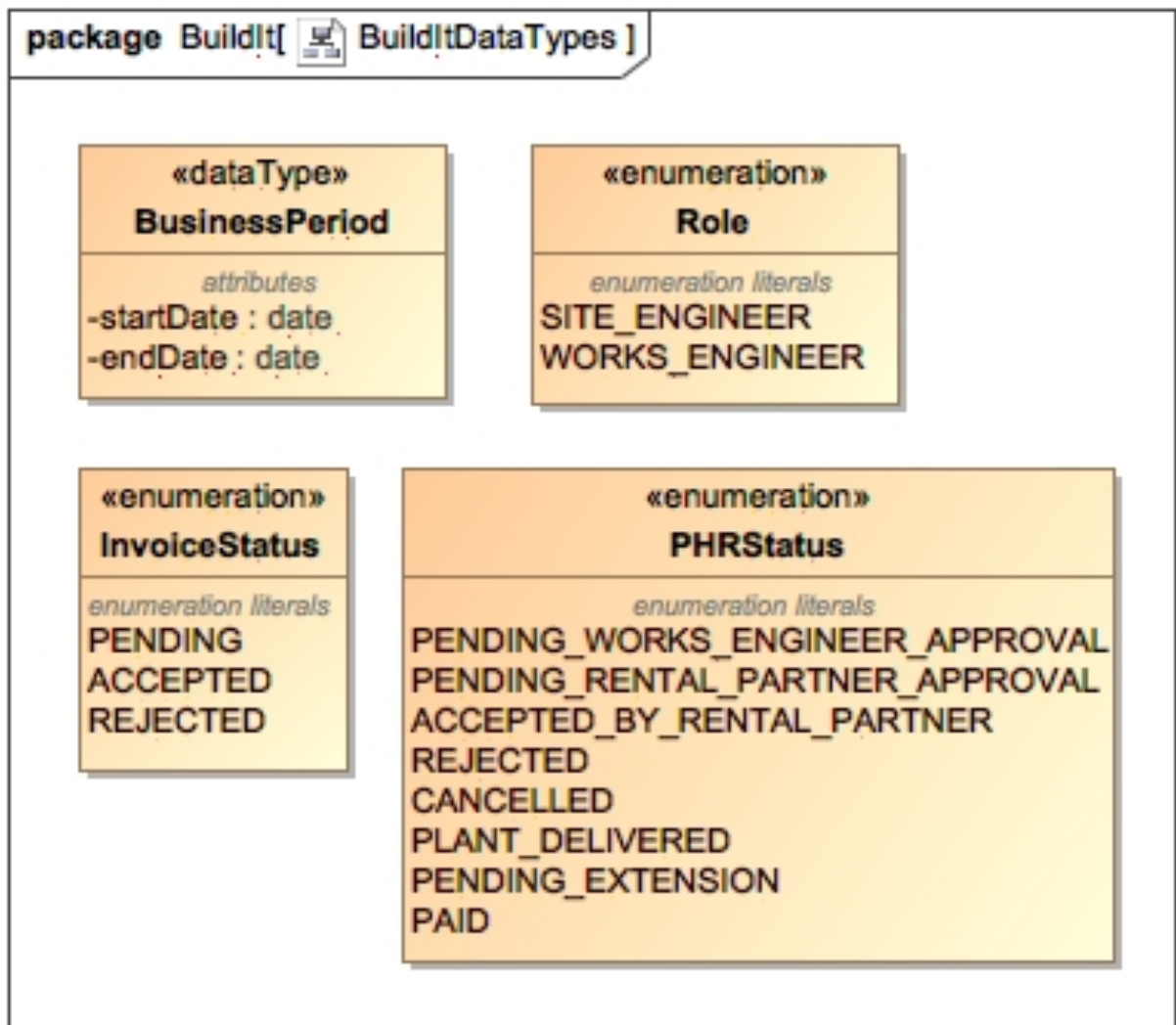PLANT_DELIVERED
PENDING_EXTENSION
PAID

Figure 2 BuildIt Data Types

Resource model

Resource model helps designing RESTful applications. It is depicted in Figure 3 for BuildIt.
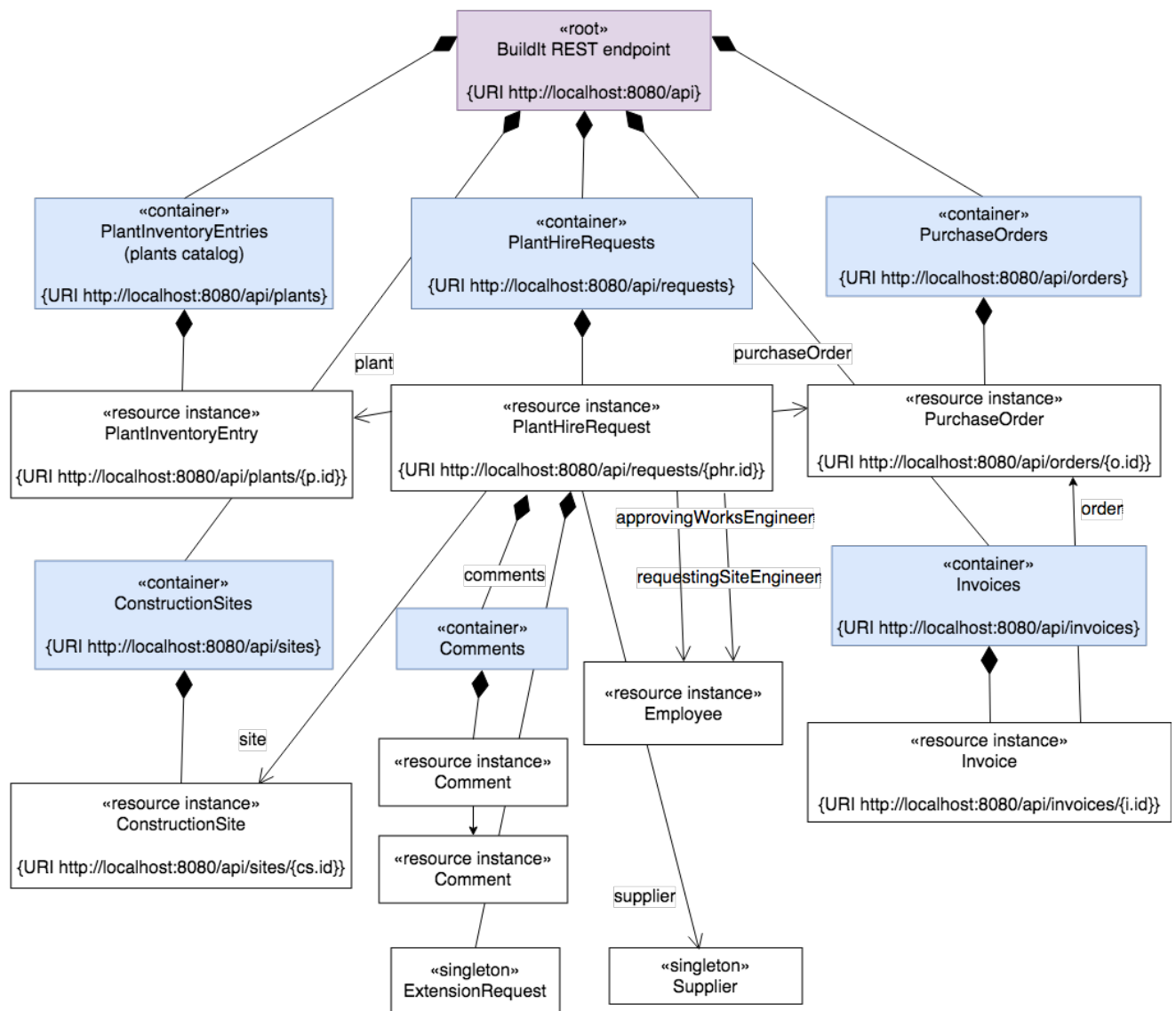

Figure 3 BuildIt resource model
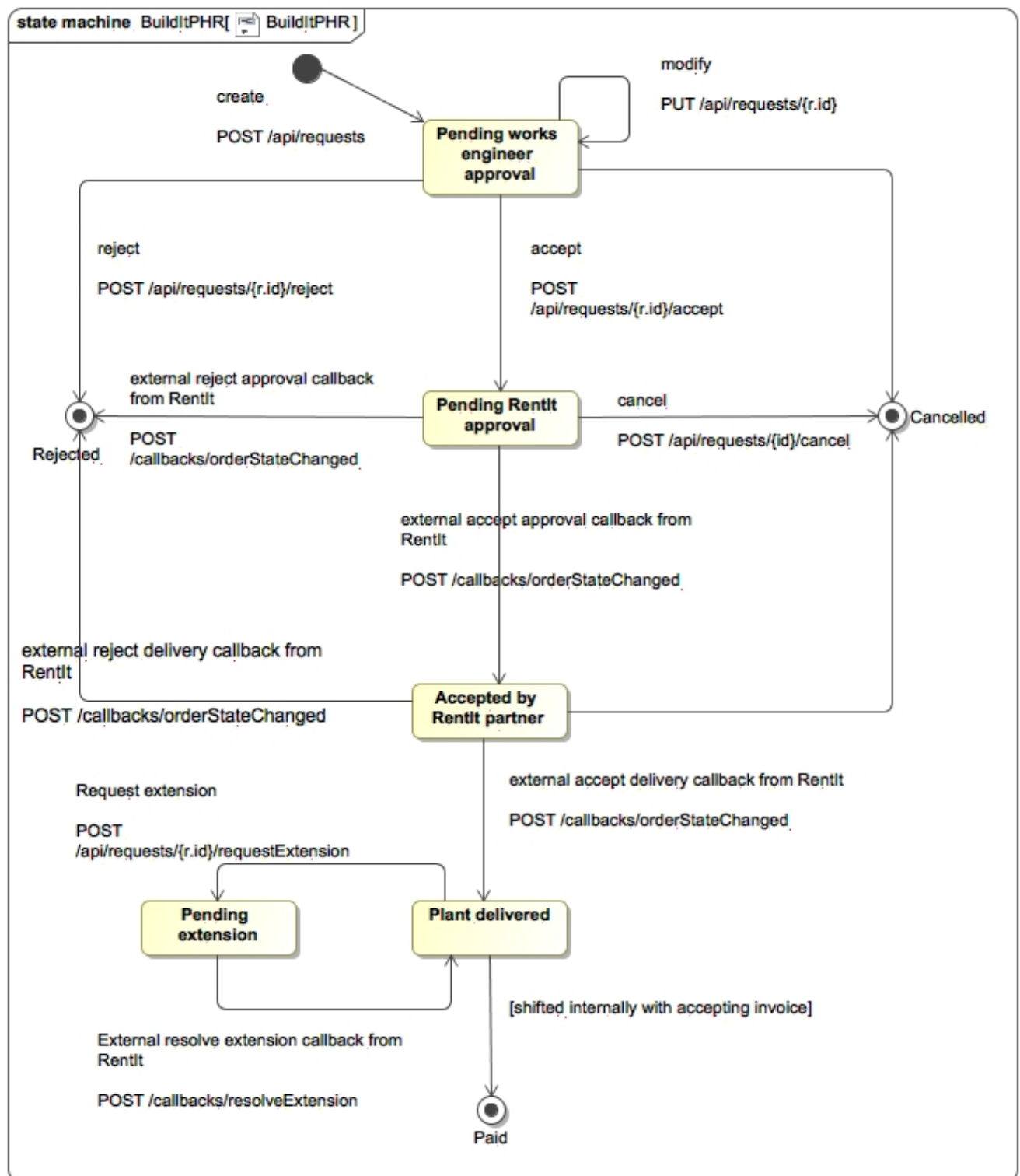
State models

Figure 4 State model for plant hire request

State model for plant hire request is depicted in Figure 4.

Notes about plant hire request state model: "Pending extension" state goes back to plant delivered state regardless of the extension being accepted or rejected. If the extension is accepted, the end date of plant hire request gets updated. No history of extension success is kept - this is left as future work. RentIt side is responsible for updating state according to delivery. This should be amended with back-up process on BuildIt side.
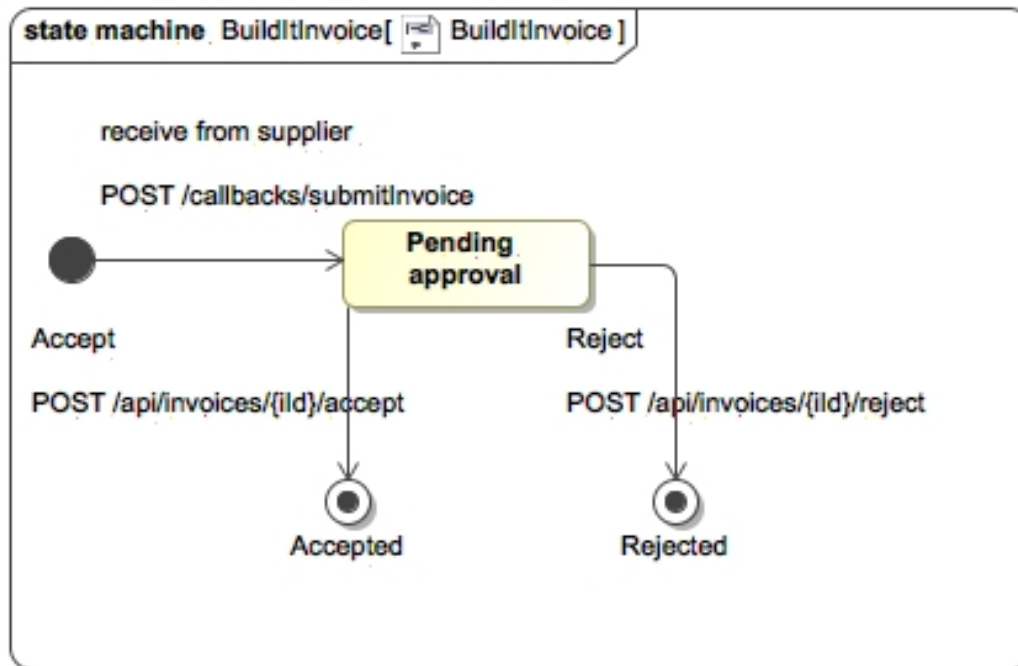

Figure 5 State model for invoice

State model for invoice is depicted in Figure 5.

Apiary blueprint

The Apiary blueprint for BuildIt system is located at:

https://buildit28.docs.apiary.io/

BuildIt API exposes the following endpoints, with their function described:
- (CC1) Create Plant Hire Request (PHR)
  - Query plants catalog - site engineer can thereafter pick a plant to rent;
  - Retrieve list of construction sites;
  - Create Plant Hire Request (PHR) – by site engineer, JSON payload contains:
    - Plant ID with corresponding supplier ID (both can be retrieved from the chosen plant's data transfer, for end-user the supplier is quite transparent in here);
    - Construction site (e.g. site engineer can select a site from drop-down in a plant hire request form)
    - Rental period
- View all PHRs / (CC4) one PHR by ID – both site and works engineer;
- (CC2, CC5) Modify PHR by ID – both engineers;
- Trigger operation on PHR:
  - (CC5) Accept/reject – by works engineer;
  - (CC3) cancel - by site engineer;
  - Comment – both engineers;
  - (CC8) Request extension – by site engineer;
- (CC7) View all Purchase Orders (PO) – both engineers;
- Calls from external partner RentIt:
  - (CC6) Receive PHR status update callback - from integrated RentIt partner;
  - (CC8) Receive extension status update callback - from integrated RentIt partner;
  - (CC9, CC10) Receive invoice - from integrated RentIt partner – it must contain the corresponding PO ID;
- (C11) Approve Invoice
  - Retrieve all invoices, optionally filtering by status – so they could be approved;
  - (C11) Retrieve PO associated with an invoice
  - (C11) Approve/reject invoice

Part II: RentIt

In this second part, we present the models for plant hiring company domain.

Domain model

First we present domain model in **Figure 6**. Identifiers are again omitted from this conceptual model.
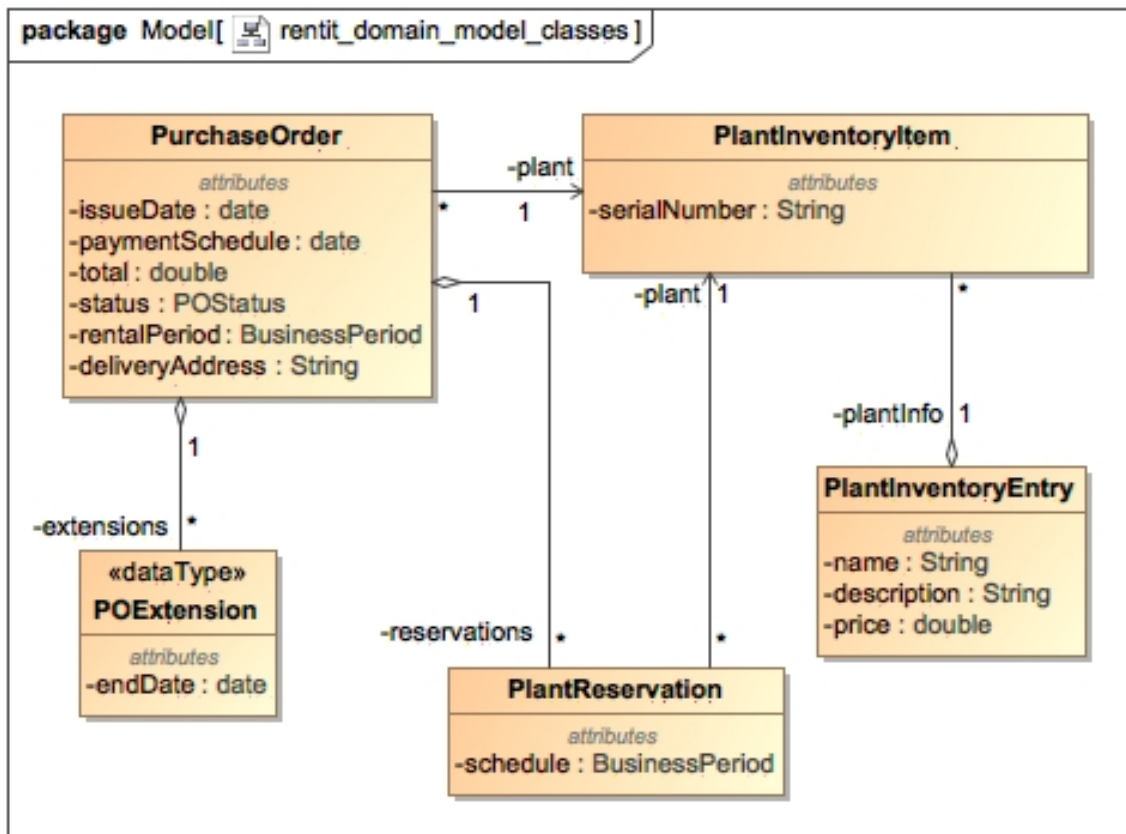

Figure 6 RentIt domain model

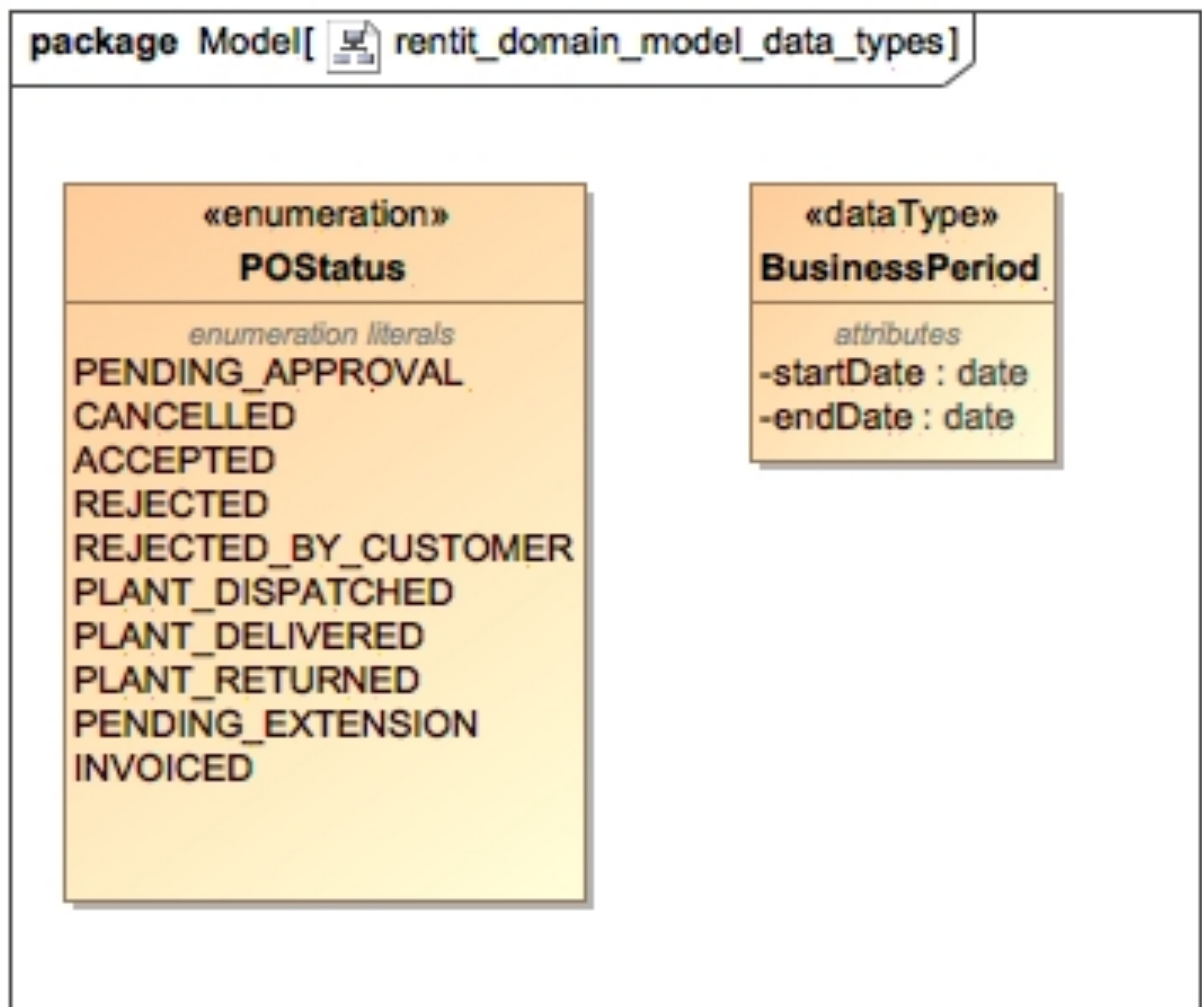The corresponding data types for RentIt are in **Figure 7**.

Figure 7 RentIt Data Types

Resource model

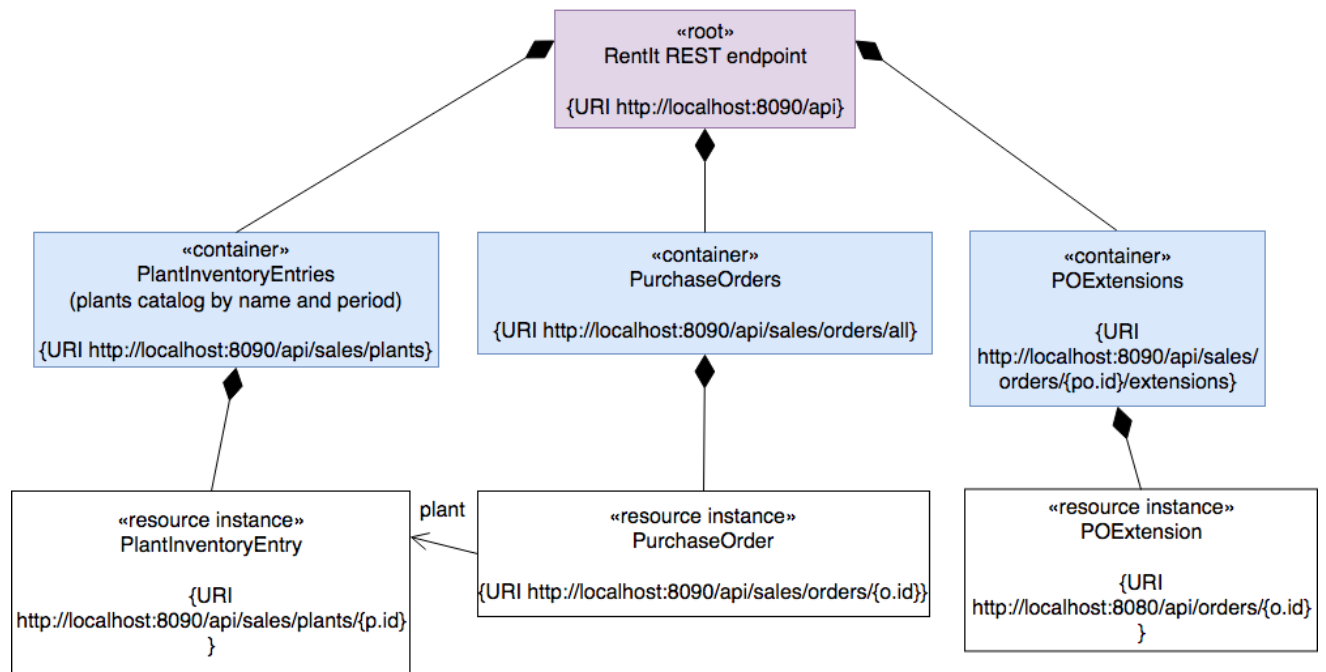Resource model is depicted in Figure 8 for RentIt.

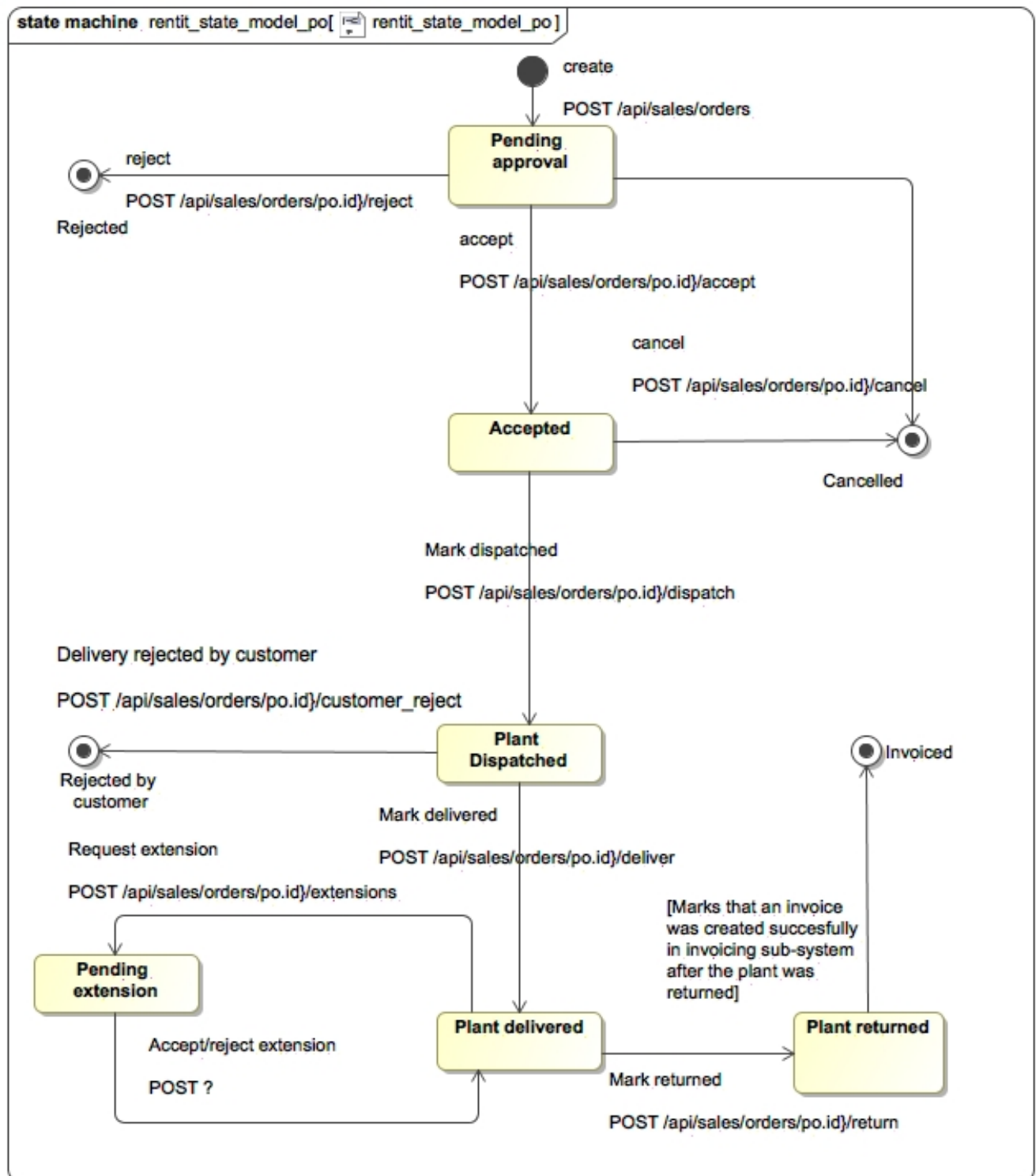**Figure 8 RentIt resource model**

State models

Figure 9 State model for purchase order

State model for purchase order is depicted in Figure 9.

Apiary blueprint

The Apiary blueprint for RentIt system is located at:

https://rentit52.docs.apiary.io

RentIt API exposes the following endpoints, with their function described:
TODO

Summary

In this report, we modeled the integrated communication between enterprise systems of a construction company, and a plant equipment rental company.