Software Requirements Specification for BeamBending: examining a beam bending under load.

Team Drasil
Jason Balaci

February 11, 2023

Contents

Revision History

Date	Version	Notes
Jan. 25, 2023	0.0.0	Template imported.
Jan. 25, 2023	0.1.0	Preliminary information added.
Jan. 25, 2023	0.1.1	Table of units added.
Jan. 25, 2023	0.1.2	Table of symbols added.
Jan. 25, 2023	0.1.3	Refining above tables.
Jan. 25, 2023	0.2.0	Refined introduction.
Jan. 25, 2023	0.3.0	Refining GSD, SSD, TMs, and IMs.
Feb. 4, 2023	0.4.0	Work on things other than the models.
Feb. 5, 2023	0.5.0	Refining models (again).
Feb. 6, 2023	0.6.0	Complete rough draft.

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	**unit**	**SI**
m	length	metre
kg	mass	gram
Pa	pressure	pascal
rad	angle	radian
N	force	newton

Note that we will also often use:

- "gigapascals" (a unit of pressure, denoted by GPa, where GPa $= 10^9$Pa),
- "kilonewtons" (a unit of force, denoted by kN, where kN $= 10^3$N), and
- "millimetres" (a unit of length, denoted by mm, where mm $= 10^{-5}$m).

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order.

symbol	**unit**	**description**
L	m	abstract length of the beam
E	GPa	abstract Young's modulus (beam material modulus of elasticity)
I	m^4	abstract moment of inertia
L_B	m	user-defined length of the beam
E_B	GPa	user-defined Young's modulus (beam material modulus of elasticity)
I_B	m^4	user-defined moment of inertia of a cross-section of the beam

not all symbols are in the table. M (Moment)? ρ (curvature) ?

$w_B(x)$	kN	user-defined definition of the force of the load applied at a specific point along the beam
x	m	distance of an arbitrary point along the beam, from the far left-side (at the pinned support)
$w(x)$	kN	hypothetical force of the load applied at a specific point along the beam
Pinned	m	1-dimensional position of the pinned support
Roller	m	1-dimensional position of the roller support
R_{Pinned}	kN	reaction of the pinned support under loaded beam
R_{Roller}	kN	reaction of the roller support under loaded beam
θ_{Pinned}	rad	slope of the angle between the beam and the pinned support under load
θ_{Roller}	rad	slope of the angle between the beam and the roller support under load
$y(x)$	mm	deflection of a hypothetical loaded beam at a specific point along it
$y_B(x)$	mm	deflection of the loaded beam at a specific point along it

1.3 Abbreviations and Acronyms

symbol	**description**
A	Assumption
BVP	Boundary Value Problem
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
BeamBending	Beam Bending
TM	Theoretical Model

1.4 Mathematical Notation

The common mathematical notation used in Canadian undergraduate-level mathematics will be used throughout this document.

2 Introduction

For any load-carrying structure, beams safely distribute the stress of the load to the foundations of the structure. Unlike flooring in residential homes, we expect the beds of industrial-strength mechanical tools and bridges to, within reason, be vertically and horizontally flexible, reacting to imposed load such that they may hold with minimal columns. *Simply supported* beams are one type of beam that are commonly found in bridges and beds of machine tools. It is important to understand how beams will react under load or else we risk damaging structures, floor bending (making inhabitants feel unsafe), or damaging loads. We may use software to analyze the beams reaction under various loading scenarios. We call the software that performs this: "BeamBending."

This document aims to develop a general scheme for understanding the reaction of a simply supported beam to imposed load under simplified conditions. This section aims to provide an overview of the Software Requirements Specification (SRS) for the "BeamBending" problem, discussing the scope and purpose of the work.

2.1 Purpose of Document

The purpose of this document is to provide the reader with a well-derived, verifiable explanation of a solution to the "beam bending" problem. The document provides sufficient information such that a related software artifact may be constructed. The produced software artifact has an "increased" degree of confidence in reliability and correctness by being traceable to and derived by these defined software requirements specifications. As such, a large focus on the development of this document is to have the domain knowledge captured and adequately codified such that the origins of fragments of code may be verified, up to development choices (e.g., language, tooling, etc.).

2.2 Scope of Requirements

The requirements analyze the "problem" (as defined in Section 4.1) and related "solution" under the assumption of the Euler-Bernoulli Beam Theory [1] in a 2-dimensional space for a prismatic beam.

2.3 Characteristics of Intended Reader

Readers of this document should at least have an understanding of:

- at least a first-year university level physics concepts, such as "force," "mass," "inertia," "elasticity," and "units,"
- first-year university level linear algebra and calculus concepts, such as derivatives, integration, continuous functions, and vectors.

However, it is preferred that users have at least a second-year university level understand of physics and calculus concepts to confidently audit the derivation of the instance models.

Should this document be used for non-trivial, non-educational purposes, this document should strictly be read and used by those appropriately licensed and well-versed in software and civil engineering.

2.4 Organization of Document

This document follows the SRS template as specified by Smith and Lai [2]. If you are already familiar with the SRS template, the author's recommended, but not required, reading order is as follows:

- Problem Description,
- Goal Statements,
- Characteristics of Intended Reader,
- Scope of Requirements,
- General System Description,
- Assumptions,
- Specific System Description, and, finally,
- Instance Models to Theoretical Models.

The other material is referential and may be read as needed.

3 General System Description

This section provides general information about the system so that the next section will be easier to digest. It identifies the interfaces between the system and its environment, describes the user characteristics, and lists the system constraints.

3.1 System Context

Figure 1 represents an abstract view of the software. The rectangular node represents the "BeamBending" software itself, whilst circular nodes represent external entities interacting with the "BeamBending" software. The abstract view is that a user would provide input information (beam and load specifications) to the software, which would use a provided, external, Boundary Value Problem (BVP) solver to process the inputs, and output the expected deflection curve.

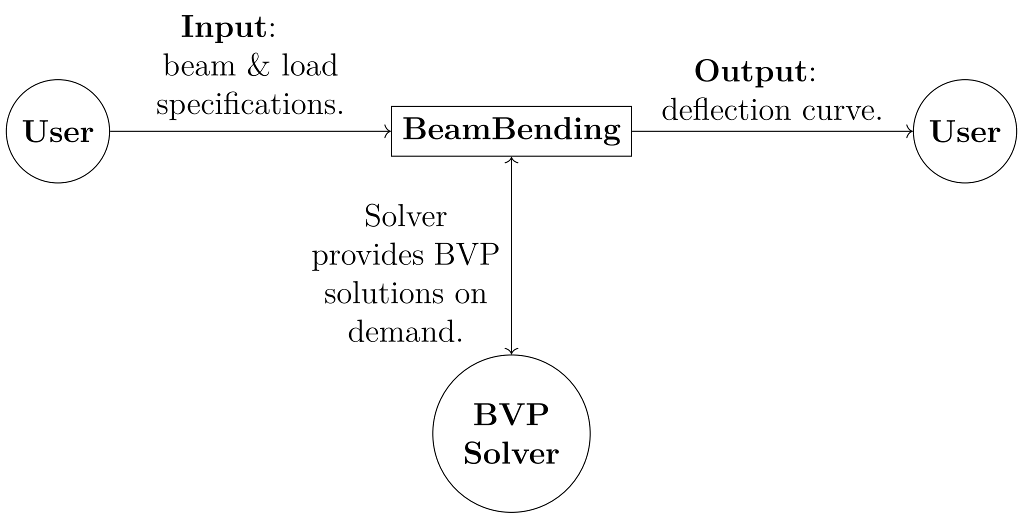

Figure 1: System Context

- User Responsibilities:
  - Providing material properties of the beam, taking required units, assumptions, and applicability of the beam into consideration.
  - Providing an explanation of the imposed load as a function of the distance from the leftward pinned support.
  - Interpret the output of the program taking this SRS document into consideration and use an audited and reliable copy of a software related to this SRS.
- BeamBending Responsibilities:

– Detect data type mismatches, such as a string of characters instead of a floating point number.
– Evaluate applicability of the input arguments to the proposed “solution” outlined in this document, such as numbers being above a certain threshold with respect to others (e.g., the number of sampling points must be a positive, non-zero, number).
– Calculate the imposed loading and resultant deflection at every desired sampling point, the angles of rotation between the beam and the respective supports, the force reactions at the supports, and the movement of the beam in the horizontal direction.

3.2 User Characteristics

The user of this software should be a civil engineer or equivalent, and have a working understanding of the deflection of beams, shear forces, bending moments, and related concepts. If applied in educational purposes, only an understanding of first-year physics and calculus is required to generally understand the program and simulate a simply-supported beam across various scenarios.

3.3 System Constraints

The final software should be built using Drasil to encode the problem and generate a problem using a BVP solver. There are no other system constraints. However, it should be noted that the usage of Drasil does impose the limitations of Drasil onto the produced software and SRS documents. One notable limitation is the inability to input general expressions into Drasil-generated programs. As such, the “user-defined” loading function, $w_B(x)$, will need to be hard-coded into both this SRS document and the software. However, changing the source for both is relatively straightforward and simple thanks to Drasil.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

BeamBending is intended to solve for the deflection a beam experiences given a load-application function and properties of the beam.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- beam: structural components intended to carry loads perpendicularly to their "long" axis,
- pinned support: a structural support that allows for rotation but neither vertical nor horizontal movement,
- roller support: a structural support that allows for rotation and horizontal movement, but not vertical movement,
- simply supported beam: a beam that has a pinned support on one end, and a roller support on the other,
- load: an applied force to a structure,
- deflection: the amount that a structural component is displaced due to deformation under load,
- shear forces: unaligned opposed forces acting on an object (e.g., shears or scissors) that can result in tearing,
- bending moment: the internal reaction of a structural element when an external force is imposed and bends it,
- Young's modulus (modulus of elasticity): the measure of lengthwise stiffness of an element as a force is applied lengthwise, and
- moment of inertia: torque required for angular acceleration.

4.1.2 Physical System Description

An "at-rest" view of the physical system of BeamBending, as shown in Figure 2, includes the following elements:

- a slender beam,
- a pinned support, and
- a roller support.

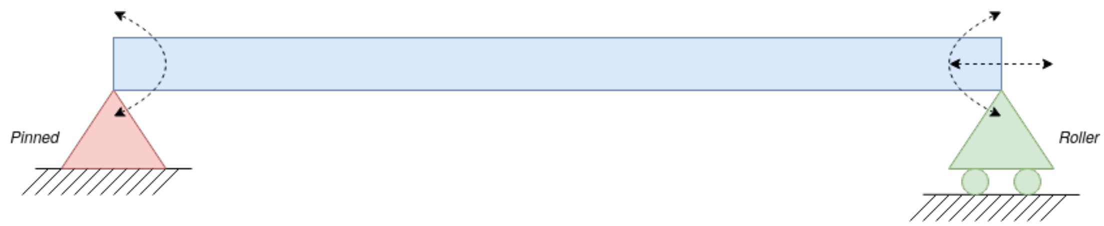

Figure 2: Diagram of a Simply Supported Beam

Figure 2 shows the physical system at equilibrium. A slender beam is supported by a pinned support and a roller support. For the sake of following convention, the pinned support is placed on the left side of the beam, while the roller support is on the right. Additionally, when thinking of distance along the beam, we think of it as a distance from the pinned support on the left-hand side. As shown in Figure 2, the pinned support allows for rotational movements about the support tip, while the roller support also allows for some horizontal movement along the roller. When the beam is loaded, the roller support allows for tensile/compressive changes in the beam (which would alter the length of the beam).

The physical system described in BeamBending examines Figure 2 under an inputted general loading function. As such, the "whole" system is as per Figure 3, containing:

PS1: a slender beam,

PS2: a pinned support,

PS3: a roller support, and

PS4: an applied load.

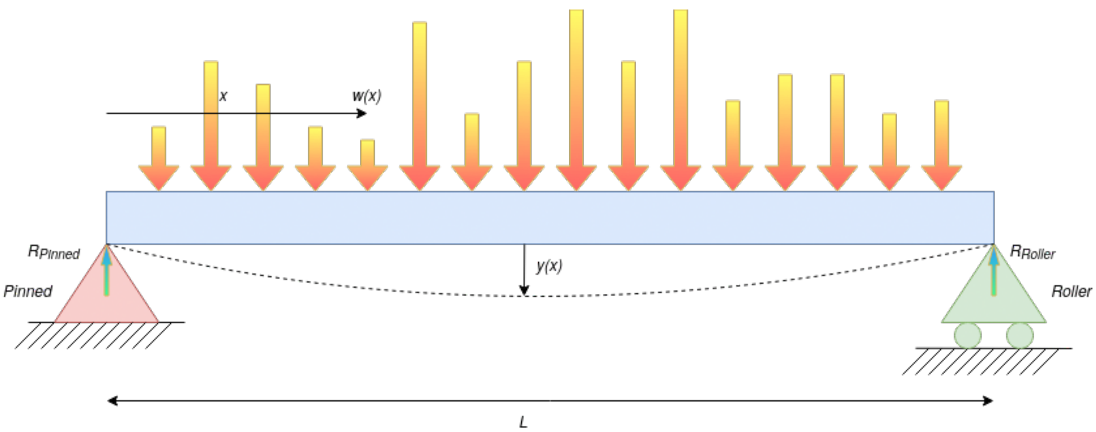

Figure 3: Diagram of a Simply Supported Beam Under Load

However, note that at the "boundaries" (the support tips), the beam may not have vertical deflection, and similarly has zero bending moment at the tips. Additionally, it should be noted that the beam is considered *determinant*.

4.1.3 Goal Statements

Given constant length, modulus of elasticity, and moment of inertia across cross-sections of the beam, and the applied load on the beam as a function from the distance of the leftward pinned support, the goal statements are:

GS1: Calculate the deflection of the beam under load.

GS2: Calculate the reaction of the supports under load.

4.2 Solution Characteristics Specification

The instance models that govern BeamBending are presented in Subsection 4.2.6. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the Theoretical Model (TM), General Definition (GD), Data Definition (DD), Instance Model (IM), or Likely Change (LC), in which the respective assumption is used.

A1: The "world" model is 2-dimensional, observing the instant a load is applied on the beam.

make this its own assumption

A2: The beam is "slender" (has a length to height ratio greater than 10 to 1).

- *The beam is prismatic.*

  A3: The beam has a uniform cross-section.

  A4: The beam is straight/flat within a reasonable tolerance.

A5: The beam has a constant ~~moment of inertia~~. Second moment of area.

A6: The beam experiences vertical linear-elastic load.

A7: The beam's modulus of elasticity is constant along the beam. ✓

should we use a symbolic constant here?

A8: Only relatively small deflections will be examined (whereby the maximum deflection is at most 10% of the beam's length).

A9: The hypothetical deflection function should belong to at least the class of functions that are differentiable 4 times on $[0, L]$.

A10: The loading function should belong to at least the class of functions that are integrable 4 times on $[0, L]$. [illegible]

4.2.2 Theoretical Models

This section focuses on the general equations and laws that BeamBending is based on.

You also have an assumption of no loading in the axial direction.

You also have an assumption that there are no point loads.

You also have an assumption on the form of $w(x)$

RefName: TM1

Label: Curvature of a Plane

Equation:

$$\frac{1}{\rho} = \frac{\frac{d^2y}{dx^2}}{(1 + (\frac{dy}{dx})^2)^{\frac{3}{2}}}$$

Description: From elementary calculus, the curvature of a plane curve at a point $Q(x, y)$ of a curve may be expressed by the above equation, where [3]:

- ρ is the radius of curvature,
- y is the vertical component of the point, and
- x is the horizontal component of the point.

Notes: None.

Source: [3]

Ref. By: TM3

Preconditions for TM1: None.

Derivation for TM1: Not Applicable

RefName: TM2

Label: Prismatic Beam Pure Bending into Arc

Equation:

$$\frac{1}{\rho} = \frac{M(x)}{EI}$$

Description: For a beam, the reciprocal of the radius of curvature is a ratio of the bending moment ($M(x)$) to the flexural rigidity (EI).

Notes: None.

Source: [3]

Ref. By: TM3

Preconditions for TM2: None.

Derivation for TM2: Not Applicable

Every ~~term~~ symbol should be defined with its units. This is where we start to run into trouble with ~~an~~ your non-base SI units

$$\frac{M\ (kN \cdot m)}{E\ \left(\frac{GN}{m^2}\right)\ I\ (m^4)}$$

These don't match, so if your units are used directly, we need a conversion factor.

$$\frac{10^3 N \cdot m}{10^9 N \cdot m^2}$$

I think it's easier if everything is in the base SI units.

RefName: TM3

Label: ODE Relationship of an Elastic Curve

Equation:

$$\frac{d^2y}{dx^2} = \frac{M(x)}{EI}$$

Description:

Notes: None.

Source: [3]

Ref. By: TM4

Preconditions for TM3: None.

Derivation for TM3: Not Applicable

Starting with TM1:

$$\frac{1}{\rho} = \frac{\frac{d^2y}{dx^2}}{(1 + (\frac{dy}{dx})^2)^{\frac{3}{2}}}$$

We may recognize that in the context of an abstract prismatic beam, the slope $\frac{dy}{dx}$ is relatively "small" (A8). Similarly, its square is also small. Considering it negligible, we may drop it:

$$\frac{1}{\rho} = \frac{d^2y}{dx^2}$$

However, substituting $\frac{1}{\rho}$ from TM2, we obtain:

$$\frac{d^2y}{dx^2} = \frac{M(x)}{EI}$$

RefName: TM4

Label: Euler-Bernoulli Beam Deflection Equation

Equation:

$$EI\frac{d^4y}{dx^4} = w(x)$$

Description: The above equation describes the relationship between a beam's deflection ($y(x)$) and the applied load ($w(x)$) for any point along the beam (x). Notably,

- $y(0) = 0$,
- $y(L) = 0$,
- $\frac{d^2}{dx^2}y(0) = 0$, and
- $\frac{d^2}{dx^2}y(L) = 0$

because the beam is simply supported (fixed at both ends implying no vertical movement), and hence the moments at the ends are 0.

Notes: None.

Source: [1, 3]

Ref. By: IM1

Preconditions for TM4: None.

Derivation for TM4: Not Applicable

When a prismatic beam supports an applied loading, ($w(x)$), its elastic curve is governed by the fourth-order ODE found by integrating the equation found in TM3 twice.

This should be another assumption

4.2.3 General Definitions

There are no general definitions.

Detailed derivation of simplified rate of change of temperature

4.2.4 Data Definitions

Number	DD1
Label	**Applied Load**
Symbol	$w_B(x)$
SI Units	~~$\mathrm{W\,m^{-2}}$~~ N/m
Equation	$w_B(x) = (x - \frac{L_B}{2})^2$
Description	$w_B(x)$ is the function describing the load applied along the beam. x is a point along the beam. L_B is the length of the beam.
Ref. By	IM1

This will be a very small load. Multiply it by a constant (the constant will be an input)

4.2.5 Data Types

The conventional data types used in undergraduate-level physics courses is sufficient. As such, the specifications define no extra data types.

4.2.6 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.4 to replace the abstract symbols in the models identified in Sections 4.2.2 and 4.2.3. Notably, GS1 is solved by IM1.

I think we should use a more general function, like a cubic

$w_B(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$

these would all be inputs

$w_B(x) = W(x - \frac{L_B}{2})^2$

Number	IM1
Label	**Beam Deflection**
Input	L_B, E_B, I_B, $w_B(x)$
Output	$y_B(x)$ as the solution to the BVP defined as: $E_B I_B \frac{d^4 y_B}{dx^4} = w_B(x)$ where $y_B(0) = 0$, $\frac{d^2 y_B}{dx^2}(0) = 0$, $y_B(L_B) = 0$, and $\frac{d^2 y_B}{dx^2}(L_B) = 0$
Description	The above output is the solution of the above defined BVP. L_B is the beam's length. E_B is the beam's modulus of elasticity (Young's modulus). I_B is the beam's moment of inertia. $w_B(x)$ is the function describing the load applied along the beam. x is a point along the beam. $y_B(x)$ is a function describing the deflection along the beam.
Sources	
Ref. By	

units?

IM1 could be a refinement of TM4 without for any beam and then IM1 is for a simply supported beam

Derivation of Beam Deflection:

This model is a direct instantiation of TM4 observed as a BVP for the beam in question as provided by user inputs (i.e., with L_B, E_B, I_B, w_B, y_B substituted in).

4.2.7 Input Data Constraints

Table 1 shows the data constraints on the input & output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise.

The specification parameters in Table 1 are listed in Table 2.

200 × 10^9 Pa

Table 1: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value	Uncertainty
L_B	$0 < L_B$	$0 < L_B$	10 m	0%
E_B	$0 < E_B$	$0 < E_B$	—	5%
I_B	$0 < I_B$	$0 < E_B$	—	5%

We might want maximum values too?

1.3e^-5 m^4

Table 2: Specification Parameter Values

Var	Value
—	—

You'll also want inputs for the distributed load function.

4.2.8 Properties of a Correct Solution

A correct solution must comply with the limitations as defined in Table 3.

One property is that for w(x) ≥ 0, y(x) ≥ 0 (for the sign convention assumed)

Table 3: Output Variables

Var	Physical Constraints
$y_B(x)$	$\forall\ x \in [0, L_B]\ .\ \lvert y_B(x)\rvert \leq \frac{L_B}{10}$ (by A8)

There is nothing in your program to actually enforce this. There is no physical reason for this to be true. Maybe you want a req. to test for this violation?

5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

R1: The inputs must be provided by the end-user according to Table 1.

R2: The inputs of BeamBending should be displayed to the user before any meaningful computation begins.

R3: The output variables should be calculated following the instance models as per Section 4.2.6.

R4: The output of the program should have a well-defined output scheme in the software design document.

5.2 Nonfunctional Requirements

NFR1: **Accuracy** The accuracy of the computed solutions should meet the level needed for civil engineering. The level of accuracy achieved by BeamBending shall be described following the procedure given in Section X of the Verification and Validation Plan.

NFR2: **Usability** The program should allow for post-processing and visualiztion of the calculated data by re-iterating all important information before termination. BeamBending is not intended to be used alone with manual analysis. The level of usability achieved by the software shall be described following the procedure given in Section X of the Verification and Validation Plan.

NFR3: **Maintainability** Through specialized tooling, the effort needed to make any likely change should be relatively straightforward and already well-understood. Additionally, any other possible change should have a clear impact outlined by the specialized tools.

NFR4: **Portability** BeamBending should be usable on at least the 3 major home operating systems (Linux, macOS, and Windows).

6 Likely Changes

LC1: The loading function is currently pre-defined and should be changed.

LC2: The units of the variables may be changed if numbers have poor readability in certain units.

LC3: The beam's configuration is likely to change across different beam needs and positions.

7 Unlikely Changes

There are no unlikely changes.

8 Traceability Matrices

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 4 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 5 shows the dependencies of instance models, requirements, and data constraints on each other. Table 6 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

	TM1	TM2	TM3	TM4	DD1	IM1
TM1						
TM2						
TM3	X	X				
TM4			X			
DD1						
IM1				X	X	

Table 4: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM1	4.2.7	R1	R2	R3	R4
IM1		X	X			
R1		X				
R2			X			
R3			X			
R4		X			X	

Table 5: Traceability Matrix Showing the Connections Between Requirements and Instance Models

9 Development Plan

The software is to be constructed using Drasil. Additionally, Drasil should be used to re-construct this SRS document. Faking the "rational design process" in Drasil allows developers to scale products against changes in the background knowledge. Drasil has extensive documentation on *how* to develop software using it and should be followed by any developer using this document.

10 Values of Auxiliary Constants

There are no auxiliary constants needed for this SRS document.

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
TM1	X	X	X							
TM2	X	X	X	X	X	X	X			
TM3	X	X	X	X	X	X	X			
TM4	X	X	X	X	X	X	X	X	X	X
DD1										
IM1	X	X	X	X	X	X	X	X	X	X
LC1									X	X
LC2										
LC3		X	X	X	X	X	X	X	X	X

Table 6: Traceability Matrix Showing the Connections Between Assumptions and Other Items

References

[1] Wikipedia contributors. *Euler-Bernoulli beam theory — Wikipedia, The Free Encyclopedia*. [Online; accessed 20-January-2023]. 2022. URL: https://en.wikipedia.org/w/index.php?oldid=1125198888 (cit. on pp. 1, 12).

[2] W. Spencer Smith and Lei Lai. "A New Requirements Template for Scientific Computing". In: *Proceedings of the First International Workshop on Situational Requirements Engineering Processes – Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP'05*. Ed. by J. Ralyté, P. Ȧgerfalk, and N. Kraiem. In conjunction with 13th IEEE International Requirements Engineering Conference. Paris, France, 2005, pp. 107–121 (cit. on p. 2).

[3] Ferdinand P. Beer and E. Russell Johnston Jr. *Mechanics of Materials*. McGraw-Hill Ryerson, 1981 (cit. on pp. 9–12).