A Technical Deliverable 1([±40%] of total  
words)  
The description of what will be presented in the technical deliverables section of the final report. This section must present and be based on a state of the art of the topics  
addressed by the technical aspects of the project.   
DO NOT PRESENT THE IMPLEMENTATION OF THE PRODUCT IN THIS SECTION   
After having given a 1 to 3 lines for the name and main goal of the product, it is advised to structure this section in sub-sections and sub-sub sections. A possible structure could  
be:

1) Context: the technical context of the product: here it should be presented the links bewteen the targeted product and the existing comparable tools and related technologies, and provided motivations for this product.

2) Functionalities: the sub-components/sub-products deduced from the list of functionalities clearly described per component for the product targeted. This section corresponds to the ”Requirements” of the product.

3) Ideas: the ideas/directions/type of solution you would propose to produce the product. In this sub-section, it must be presented an overview/high level understanding of what is expected to be the product produced.

Some wireframe / prototype of the GUI can be used to the solutions for the product services/functionalities. Libraries used and general code overview can be explained. This  
section corresponds to the ”design” of the product.  
In the three sub-sections proposed it must be provided a commented use of the relevant references on which the work is proposed to be based on.

1-3lines [62]

As mentioned previously, the technical deliverable of this project is a tool, named “BIM bluepring manager “, upon which a standardized set of data is transformed into a BIM construction plan. This tool can be used to either create an entirely new BIM model, or, to alter a pre-existing BIM model with the desired properties mentioned in the input of the user.

1. Context: the technical context of the product: here it should be presented the links bewteen the targeted product and the existing comparable tools and related technologies, and provided motivations for this product. [325]

BIM blueprint manager or BIMbm, is a tool that, as the name implies, manages blueprints. These blueprints refer to, but not exclusively, construction plans.

These construction plans are digitalized into a BIM format. BIM is widely used as digitalized blueprints, and in the context of this project, BIM construction blueprints are handled.

Various tools and technologies are compatible with BIM models.

Revit, is the tool of choice to handle BIM for this project. Revit renders BIM models, allows for manipulation, visualization and many others. Revit is a broad software, and it has many subtools which are compatible with it, one of them being Dynamo for Revit.

Dynamo for Revit, or Dynamo for short, is a pluggin for revit and a library for dynamo Nodes \*|1 .

[Different] from Revit, which to my knowledge, only allows for in place modification of BIM models, Dynamo allows for the creation of scripts that can modify a BIM upon being executed.

Other worth mentioning tools, that, although not fundamental, do help in the completion of this task. Python 2 and Python 3 are compatible with Dynamo and will be used whenever some functionality is required that is not already provided by Dynamo’s [provided default] dynamo Nodes

Lastly, BIMvision, a visionalisation tool for BIM models is to be used throughout the making of BIMbm. Although revit offers some basic visualization features of BIMs, BIMvision is a tool specifically designed for that and has more functionalities, so, it is preferred over Revit. It allows the user and [coder] to see the changes made by BIMbm in real time clearer.

\*|1 https://github.com/DynamoDS/DynamoRevit

2) Functionalities: the sub-components/sub-products deduced from the list of functionalities clearly described per component for the product targeted. This section corresponds to the ”Requirements” of the product. [607]

The goal for BIMbm is to allow the user to alter or create BIMs.

Dynamo is the main tool used for coding BIMbm because of the utility of the scripts that can be run to achieve a specific desired task.

Standardized Input :

The script, must receive a standardized input that contains the data to be modified.

A good example of standardized input data that is compatible with the desired final version of BIMbm is my BSP2 tool.

BSP2’s technical tool, extracts structural data from a construction’s BIM and outputs it into an excel file.

This data is combined in one single excel filed (.xlxs). The data is structured and organized into excel sheets. Each excel sheet refers to a single components type withing the structure, e.g. walls. In each sheet, columns refer to different properties of the components e.g. Length. As can be seen in the attachment below.

A screenshot of a computer

Description automatically generated

BSPbm’s Requirements:

The BSPbm tool would alter values in BIM models according to the user’s instructions.

The values to be changed will be input in an interface.

The user, when changing values, will not need to keep in mind inter relationships of walls. Meaning that, if the user changes the data pertaining to an arbitrary Wall A into Wall A’; Consequently, Wall B that is originally connected to Wall A would then automatically have its starting point in the new end point of wall A’. Resulting in a conflict free and more convenient way to change data.

The tool will house similar features for other BIM components: walls, rooms, windows, and doors.

It will then generate a new BIM model with the altered data.

The tool can alter or create entirely new models.

3) Ideas: the ideas/directions/type of solution you would propose to produce the product. In this sub-section, it must be presented an overview/high level understanding of what is expected to be the product produced. [771]

The first functionality, and the only one to be implemented until now, is to create BIM files from available data. Suppose, the goal is to create a simple 2D planar view of a building, with a fence around it, such as in the picture below.

A diagram of a house

Description automatically generated

A screenshot of a computer

Description automatically generatedThe premise is to input the information relevant to the creation of the building in some sort of storage that can handle the data. For the earlier builds of BIMbm, the storage of input data is excell .xlxs, later on it most likely will be .CSV format.

The data represents points with x and y coordinates in the A and B columns respectively.

An earlier version of the tool BIMbm creates a simple BIM from an excel sheet filled with data.

A drawing of a diagram

Description automatically generated

This is the output from the excel sheet of data.

This method can be applied to bigger and more complex BIMs. Furthermore, this concept can be expanded upon. For instance, we can give these connected dots, wall properties.

A blue cube with a square top

Description automatically generated

The final product is supposed to allow the end user to change any existing BIM file. Walls can be changed without disrupting the file, meaning that the integrity of the building is preserved, and subsidiary changes are made to accommodate for a changed wall. Doors and windows are also planned to be modifiable. A group of walls can be used to create a room, and existing rooms will be retained as a room when the walls are modified. All these functionalities are expected to be in a plugin dropdown list, whereby the user can click a plugin and it will run certain functionalities. Here is an abstract example of this concept.

A screenshot of a computer

Description automatically generated

Some wireframe / prototype of the GUI can be used to the solutions for the product services/functionalities. Libraries used and general code overview can be explained. This  
section corresponds to the ”design” of the product.  
In the three sub-sections proposed it must be provided a commented use of the relevant references on which the work is proposed to be based on.