SoftwareX 22 (2023) 101398

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

Precise: A web-based 3D visualization and manipulation application for surgical planning of tumour resection

Angela Busato [a], Michaela Servi [b,*], Yary Volpe [b], Rocco Furferi [b], Maurizio Scorianz [a]

[a] *Careggi Hospital of Florence, viale Pieraccini, 50139, Florence, Italy*
[b] *Department of Industrial Engineering, University of Florence, Via Santa Marta 3, 50139, Florence, Italy*

ARTICLE INFO

Article history:
Received 31 December 2022
Received in revised form 27 April 2023
Accepted 2 May 2023

Keywords:
Surgical planning
Rapid prototyping
Medical application
3D manipulation

ABSTRACT

This paper presents a collaborative platform developed to allow the communication between surgeons and engineers in the process of designing patient-specific surgical instruments. To date, only a few applications are available to collaboratively create surgical instruments from medical 3D models, mostly dedicated to expert CAD modelers. This makes the preoperative planning process time-consuming and inefficient limiting the usability of applications and making planning difficult and inaccurate. Accordingly, we propose a solution in the form of a web-based, interactive, extendable, 3D navigation and manipulation application, called *Precise*, which does not require client installation. *Precise* is a lightweight, high-performance application built to provide easy-to-use, powerful, on-demand visualization and manipulation of 3D images, implemented using open-source libraries.

Code metadata

Current code version	v1
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-22-00432
Permanent link to reproducible capsule	
Legal code license	GNU General Public License (GPL)
Code versioning system used	git
Software code languages, tools and services used	JavaScript, SQL
Compilation requirements, operating environments and dependencies	The *package.json* file is provided
If available, link to developer documentation/manual	
Support email for questions	michaela.servi@unifi.it, busato.angela@gmail.com

1. Motivation and significance

The software application described in this work was developed in the context of the project Precise (https://www.preciseproject.it/) whose main aim is to research, integrate, and validate 3D technologies in oncology practice. The project is coordinated by the Department of Industrial Engineering of the University of Florence in collaboration with the Careggi Hospital, Meyer Children's Hospital, and the Azienda Ospedaliera Universitaria Senese.

The project leverages the use of Additive Manufacturing (AM) and Reverse Engineering (RE) technologies for the customization of surgical treatment. Through the use of RE techniques to digitize anatomy and AM techniques to fabricate customized devices, in fact, it is possible to define personalized procedures and surgical instruments that take into account the anatomical uniqueness of the patient and significantly improve surgical outcomes and surgical time [1,2].

Commonly, Patient-Specific Instruments (PSI) are modeled by engineers and validated by surgeons in an iterative framework, which can be very time-consuming as engineers are not aware of all the clinical critical aspects to take into account. Specifically, the process of modeling surgical guides for the removal of bone tumours starts with diagnostic imaging segmentation, which is crucial to allow a precise 3D reconstruction of the tumour, bone, and surrounding tissues. Only through accurate identification of the lesion and valid planning of bone resections with adequate safe margins, it is possible to spare vital structures and, whenever possible, joints. The design phase is driven by these specifications,

* Corresponding author.
E-mail addresses: busato.angela@gmail.com (Angela Busato), michaela.servi@unifi.it (Michaela Servi), yary.volpe@unifi.it (Yary Volpe), rocco.furferi@unifi.it (Rocco Furferi), maurizio@scorianz.com (Maurizio Scorianz).

https://doi.org/10.1016/j.softx.2023.101398
2352-7110/

Table 1
Comparison of 3D manipulation software.

Name	Software type	Files extensions	OS	Price	Maintained
Blender [8]	Desktop	STL, OBJ, PLY	Windows/MacOS	Free	Yes
ZBrush [9]	Desktop	STL, OBJ, VRML	Windows/MacOS	42.70€/mo	Yes
Sculptris [10]	Desktop	STL, OBJ	Windows/MacOS	Free	Yes
Forger [11]	Mobile App	OBJ, C4D	iPadOS	1.99€/mo	Yes
Armorpaint [12]	Desktop	STL, OBJ, GLB	Windows/MacOS	19€	Yes
3DCoat [13]	Desktop	STL, OBJ, PLY	Windows/MacOS	19.85€/mo	Yes
Polybrush [14]	Desktop	OBJ	Windows	185€/mo	Yes
BrainBrowser Surface Viewer [15]	Web App	OBJ, JSON	All	Free	Yes
Mecuris [16]	Web App	STL, OBJ	All	Free/Pay plans	Yes

provided by surgeons and, when ready, the guides can be manufactured and sterilized. This process can suffer from the lack of communication between surgeons and engineers regarding the surgical approach and design constraints, resulting in a delayed process that affects all involved stakeholders, i.e. the surgeon, the CAD modeler and the patient.

In this context, it has emerged that there is a need to have a software that visualizes and manipulates 3D data that can be used by non-specialized users in a collaborative framework. When considering the domain of preoperative planning software, there are several tools available to simulate surgical procedures and bone resections. Currently the market offers several software packages specifically designed for preoperative orthopedic planning that allow to simulate many surgical steps. The most popular planning software packages are: Mimics [3], Sectra [4], mediCAD [5], PeekMed [6], TraumaCAD [7]. These applications typically require the user to be well trained and have a deep knowledge of the digital environment. Looking for applications in which surgeons can manipulate 3D data independently, we can evaluate programs such as Blender [8], ZBrush [9], Sculptris [10], Forger [11], Armopaint [12], 3Dcoat [13], Polybrush [14]. These software are typically used for rendering, modeling, sculpting, animating, and video editing which provide features for users that want to create models for gaming, design, and story animation. As a consequence, the interface frequently has too many functionalities that make the application tricky to use for non-medical users and prevent the creation of a clear virtual communication channel between surgeons and engineers. On the other hand, programs like BrainBrowser Surface Viewer [15] and Mecuris [16] were specifically developed for use in the medical field with particular attention to the needs of doctors. In such software packages, the interface is more user-friendly with respect to the previously described applications since the defined features are dedicated to medical needs. Moreover, these applications are web-based software, in fact, the users can visualize and manipulate the 3D model in complete autonomy, and work on it without the supervision of engineers or other users. This gives the surgeon the freedom to analyze and study the best solution for pre-operative planning and in a second moment to share that solution with the engineers.

In Table 1, we provide a comparison of the most used software considering software type, extension of handled files, price, operative system, and maintenance status.

The adoption of an information-sharing platform is the first and most important step toward a proficient concurrent design. Unfortunately, these solutions are strongly specialized for a particular medical field, making it challenging to add features addressing other, different, clinical applications. In addition, it is the authors' opinion that despite these applications are designed to be user-friendly, they can still be difficult to be used by non-expert users.

In this context, the aim of this work is to implement a unique software that surgeons could access at any moment to create and visualize 3D images and manipulate them to define the best position for the PSI and make the correct inputs to be subsequently elaborated by CAD modelers to create the PSI. Our solution consists of a web application that does not require special user computational requirements or client installation. This entails that users will be able to collaborate, share, store, and access projects at any time. *Precise* is designed with an intuitive, easy-to-use, and user-friendly interface to be used both by specialized and non-specialized users.

2. Software description

The *Precise* application was developed to enable the preoperative planning of a tumour resection. The basic idea is to define on the 3D digital model of the anatomy the best tumour resection strategy. This is accomplished, as explained below, through the insertion of a number of cutting planes which determine the correct resection paths; such planes are subsequently used to model the surgical guide. The main features of the application were discussed with a team of surgeons operating at the Careggi Hospital. Identified important features were: the possibility to freely add and manipulate planes and use them to simulate tumour resection (accordingly cut the 3D model), add 3D models of screws (with the aim of correctly planning their insertion during surgery), perform measurements on the 3D model. Other requirements, functional to the effective use of the software, were: the possibility to simultaneously import a group of files in STL format representing a specific anatomical region affected by tumour, to change the color of the entire file (to identify different parts of the body), to paint only specific regions manually indicated by the user (i.e. to highlight sensitive areas of the anatomy), offset an element (with specific reference to the tumour area, so that a safety margin could be considered in the design). Finally, it was requested both to incorporate a section dedicated to communication between engineers and physicians and to guarantee the software to be cross-platform and compatible with both computers and tablets.

2.1. Software architecture

Fig. 2 shows the architecture layers of the application. The web application container is divided into two parts: frontend and backend. The frontend contains the UI layer that defines all the required functionalities for 3D data elaboration and 3D rendering, while the backend contains the server and the database. The user connects to the application via frontend, which in turn connects to the backend via APIs. The used libraries, described below, allow as much as possible the processing on client-side, thus minimizing the effect of network lag on its usage and eliminating the need for browser plugins.

For what concerns the frontend, the following elements were designed: **HTML5 Canvas element**, a high-performance, scriptable 2D drawing surface. The canvas element provides both a 2D drawing context exposing an API for drawing basic 2D shapes and images and a WebGL context exposing an API for high-performance 3D graphics. The key aspect of the 3D API used

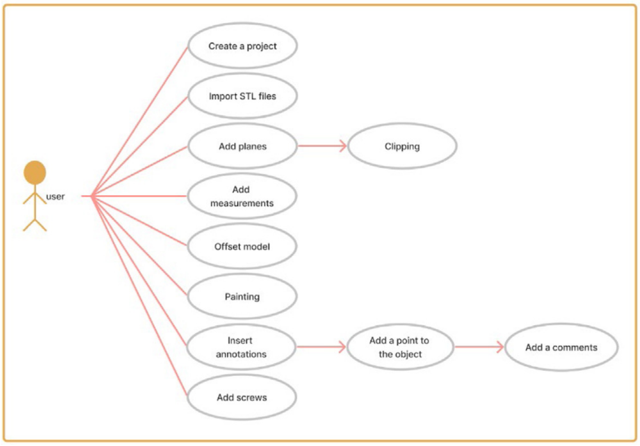

Fig. 1. Use cases defined in collaboration with medical staff.

by *Precise* is a set of functions for pixel-level image processing, manipulations, and rendering. **WebGL** [17], a low-level JavaScript graphics API that makes use of the HTML5 Canvas element to provide web pages with access to the GPU of the client computer. *Precise* makes use of WebGL through the Three.js JavaScript library. **Three.js** [18], a lightweight, cross-browser, open-source JavaScript library developed to abstract much of the complexity of using the WebGL API directly. The power of this library is that it allows the creation of a graphical processing unit using JavaScript as part of a website without relying on proprietary browser plugins. *Precise* uses this library to elaborate the 3D scene directly in the browser.

The other tools involved in the application are **React** [19], a JavaScript library that designs simple views for each state of the application, for building the user interface; **Node.js** [20], an open-source, cross-platform runtime environment and library that is used for running web applications outside the client's browser; and **PostgreSQL** [21], as an object-relational database system to store information about users and projects.

2.2. Software functionalities

The aforementioned tools were used for the development of functionality to achieve the implementation of the use cases listed in Fig. 1, some of them were implemented using features provided by the Three.js library, such as importing STL objects, moving objects in space, and inserting known objects (such as screws or points) into the scene. For others features, open-source APIs were integrated, such as for painting the faces of an object [22] or for measurements in space [23]. Finally, for a few but important features it was necessary to define dedicated procedures, which are described below.

Global alignment. An important first feature involve the alignment of all imported objects with the center of the global reference system. To accomplish this, the program saves in a centralized state the center of gravity of the first imported object and its displacement with respect to the global reference system. This displacement is then applied to all elements managed in the project, both in the current session and in future sessions (even those imported at the same time). This step is essential to ensure easy navigation of the 3D model for the medical personnel, and it is almost never found in commercially available software. To avoid future misalignment of exported models or PSI, the software, when any element is exported, removes the displacement using the translation saved in the central state. This also applies to subsequently inserted elements such as planes and screws.

Clipping. This is a very important feature as it is used to define the tumour resection paths used for modeling the surgical guide. The clipping feature refers to either clipping the object with a single plane or with multiple planes, and in both cases the clipping must be negated. To implement clipping with a single plane and its negation, the corresponding functionality provided by the Three.js library was used. On the other hand, to implement clipping with multiple planes, a dedicated routine was implemented. To align the normal of all inserted planes the implemented procedure performs two steps: (1) calculation of the barycenter of the planes' centers (denoted by c in Fig. 3); (2) check for each plane whether the normal faces the barycenter, and if not, reverse the normal, thus aligning all the normals toward the planes' barycenters.

Offset. In this work, a dedicated offset procedure was developed, since it is currently lacking in the Three.js library. The procedure, implemented in JavaScript, creates the mesh of a selected STL model by offsetting faces and vertices by a predefined value, entered by the user. To define the position of vertices and faces in the offsetted model, the vertices of the original model are saved in a hash table (this allows optimizing the performance of the algorithm). To define the direction of displacement of the vertices, the normals of the faces insisting on that vertex are averaged, then the vertex is moved along the average direction by the amount entered by the user, and finally the faces and their normals are recalculated based on the displaced vertices. Given the purpose of use of the program, it was not deemed

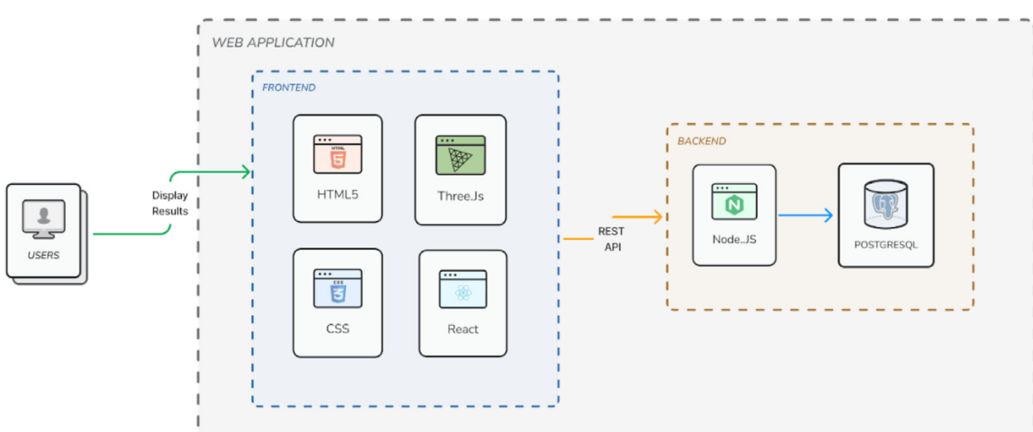

Fig. 2. Application architecture.

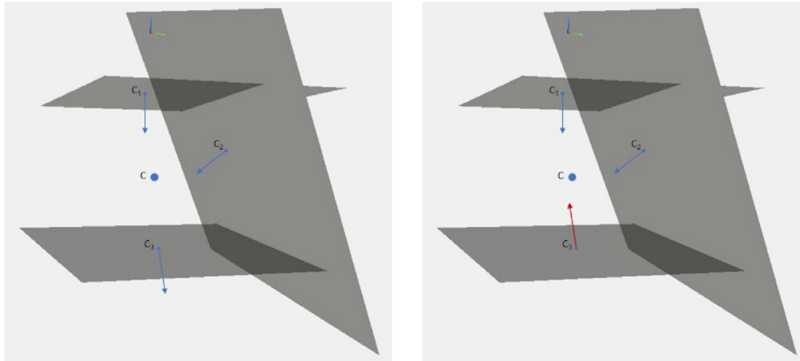

Fig. 3. Normal alignment procedure for clipping with multiple clipping planes.

necessary to correct any intersecting faces. The offset feature can be applied to each element of the scene (in both negative and positive directions) and the transparency of the offset model can be adjusted to visualize the intersections between elements.

2.3. Usability of the software

The application was designed to be used independently by medical staff. Therefore, it was of critical importance to assess the actual acceptance and user-friendliness through usability testing. To evaluate the usability, the procedure for planning the resection paths was divided in the following two tasks, evaluated individually:

Task 1- the first task deals with the actual planning phase, i.e., the use of all the functionalities necessary to define the resection path on which to design the surgical guide. Such operations include: file import, model navigation/motion, saving the project, inserting planes and performing clipping, inserting screws, offset of models, and performing measurements.

Task 2- the second task refers to the functionalities required for communication between doctors and engineers. These operations include annotation insertion and painting.

According to the ISO 9126 series of standards (the most extensive software quality model developed to date [24]), usability is interpreted as "the extent to which a product can be used by specific users to achieve specific goals with effectiveness, efficiency and satisfaction in a specific context of use". The evaluation of each metric was carried out as follows:

Effectiveness is calculated by measuring the completion rate, assigning a binary value of '1' if the test participant succeeds in completing a task and '0' otherwise.

$$Effectiveness = \frac{number\ of\ tasks\ completed\ successfully}{total\ number\ of\ tasks\ undertaken}$$

Efficiency is measured in terms of task execution time, that is, the time taken by users to achieve goals.

$$Time\ Based\ Efficiency = \frac{\sum_{J=1}^{R}\sum_{i=1}^{N}\frac{n_{ij}}{t_{ij}}}{NR}$$

Where N is the total number of tasks, R is the number of users, n_{ij} is the result of task i by user j, t_{ij} is the time taken by user j to complete task i.

Overall satisfaction was measured through the System Usability Scale (SUS) questionnaire [25]. The SUS consists of a 10-item questionnaire with five response options, ranging from Strongly Agree to Strongly Disagree. To interpret the results of the questionnaire, participants' scores for each question are summed and then multiplied by 2.5 to convert the original scores from 0–40 to 0–100. Although the scores are 0–100, they do not represent percentages and should be considered only in terms of percentile rankings: values below 51 are to be considered strongly disagree, 68 represents the threshold of sufficiency, and results above 80.3 are considered optimal.

Table 2
Effectiveness results for each user.

	User#1	*User#2*	*User#3*	*User#4*	*User#5*
Effectiveness	100%	100%	100%	50%	100%

Table 3
Results of the SEQ questionnaire.

	User#1	*User#2*	*User#3*	*User#4*	*User#5*
AVG SEQ score	6	7	5	4	6
Task with lower score	Task 1	Task 1	Task 1	Task 1	Task 1

Finally, *Satisfaction* is measured through standardized satisfaction questionnaires: the SEQ (Single Ease Question) questionnaire, which uses a seven point rating scale and is reliable, sensitive, valid and easy to respond [26]. SEQ was administered after each task.

Since it is common practice to test usability on five users [27], five orthopedic surgeons from Careggi Hospital were chosen for this study. A preliminary meeting was held to explain the functionality of the software to the participants and to give them a chance to familiarize themselves with the procedure. After this preliminary phase, the participants' tests were individually programmed and were performed by the participants without the help of observers.

Below are shown the results of the usability metrics described above. As shown in Table 2 effectiveness scores are all above 80%.

Efficiency, which is the speed of work with the product, is typically compared with the Expert Efficiency, the highest possible speed of work. In this work, during the efficiency evaluation, it was noted that these values are excessively case-dependent, and therefore it was chosen not to report them as they were not significant.

Finally, the tested software achieved an average SUS result of 91, declaring a high level of user satisfaction.

SEQ results are reported as the average SEQ value obtained by each user (see Table 3). Considering that satisfaction results typically range between 4.8 and 5.1 [28], results demonstrate a general high score of satisfaction by single users. The most challenging task for users with no 3D modeling expertise has been task 1.

3. Illustrative examples

As mentioned above, the purpose of the *Precise* application is to visualize and manipulate 3D data through a web browser

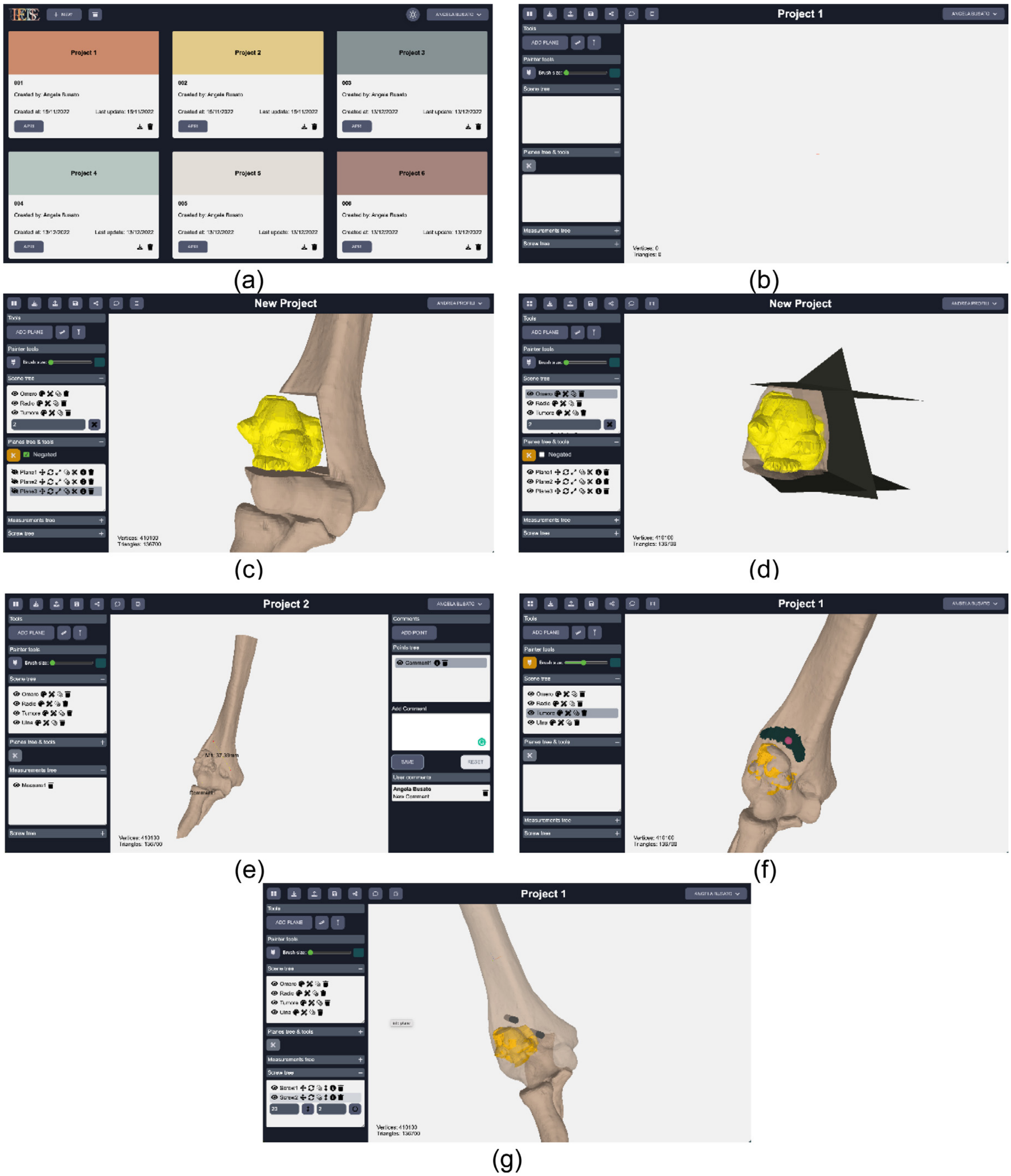

Fig. 4. Example of using the *Precise* application to plan the resection of a tumour (yellow element) on the humerus (beige element): (a) initial page showing all projects created by the user; (b) example of the project when it is first created; (c) use of the clipping function with multiple planes to simulate tumour resection; and (d) reversal of clipping direction, in the figure the tumour (in yellow) has first been offset; (e) example of measuring the distance between two points and, on the right, the insertion of an annotation; (f) example of drawing with brush on the mesh; (g) insertion of screws to indicate where the cutting guide will be fixed on the bone. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

to study, create and share preoperative planning for tumour resection. Through the use of the technologies listed above, it was possible to achieve these goals.

When the user authenticates to the application, the first page he/she encounters is the projects page (see Fig. 4a). This page shows a list of projects created by the user or those that have been assigned to him. The user can create, delete, archive or open a project. When a project is accessed, the application opens the dedicated editor page (Fig. 4b). This page contains the main functionalities of the application with all the tools defined in the requirement definition phase. In particular, the editor page is divided into four sections as follows. The header contains a list of buttons related to file manipulation (i.e., import, export, save), a button to share the project and the user profile settings. The sidebar on the left contains the main tools of the application to handle the objects in the scene. Centrally the page is dedicated to 3D navigation, and the sidebar on the right contains the annotation functionality.

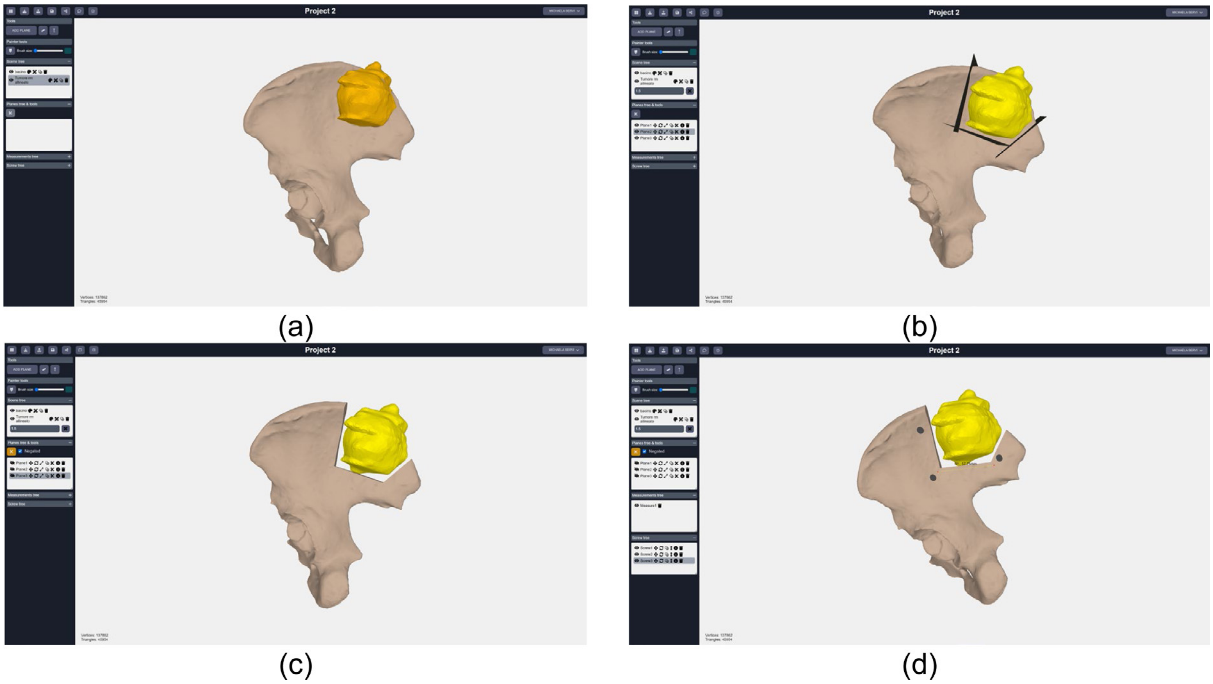

Fig. 5. Example of using the *Precise* application to plan resection of a tumour (yellow element) on the pelvis (beige element): (a) initial import of objects; (b) placement of cutting planes after offsetting the tumour; (c) use of cutting planes to verify complete resection of the tumour; (d) insertion of screws to indicate where the cutting guide will be fixed on the bone. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The user experience while utilizing the application is described below in order to make clear how the use cases were implemented.

Create a project. The user can create a new project by clicking on the "New Project" button. The system opens a modal with two input boxes to insert the name of the project and the code of the patient associated with the project.

Import of STL files. To start working with the application, the user must import the files from her/his library. Using the "Import" button the user can import one or more files in STL format. As said, once the file (or a set of files) has been successfully uploaded, the implemented software automatically aligns it with the global reference system to facilitate 3D navigation.

The sidebar on the left shows the object (or objects) with a set of tools: change visibility, delete, change transparency, change color and offset model. The latter is a function that allows the user to create a new mesh by offsetting the original vertices and faces. The user can enter a number that indicates in mm how much to increase the size of the object.

Add planes to the scene. This feature is one of the most relevant for the devised application since it simulates where the surgeon hypothetically will resect the tumour. The user can click on the button "Add plane", in the left sidebar to add a plane object to the scene. The user can interact with planes with the control buttons and manipulate the objects to position it in the scene, specifically, the user can translate the plane along the three axes by activating the translation handler, or rotate it with the rotation handler. Plane objects have several tools at their disposal: the transparency tool, an information button that gives the normal and the center of the plane and the clipping button.

Clipping. This function is a nodal point for tumour resection planning as it allows to simulate a cut and observe the repercussions on the anatomical elements. There are two ways of using the cutting function: using a single plane or using multiple planes. Fig. 4c shows an example of a cut obtainable by means of multiple planes, performed after offsetting the tumour of 2 mm to obtain a safety margin in the positioning of planes. As can be seen in Fig. 4d it is also possible to reverse the cut.

Add measurements. The user can activate measurements by clicking on the button with the ruler icon. Once it is activated, the user can subsequently select two points on the mesh and their Euclidean difference is automatically calculated and shown on the visualizer (Fig. 4e).

Painting. This functionality allows the user to color the mesh at specified faces (Fig. 4f). It is useful for highlighting some parts of the mesh, to indicate an important point that should not be changed or touched during surgery. The painting function is activated by clicking on the button with a brush icon. It is possible to change the size and color of the brush with the dedicated tool.

Insert annotations. The "Comments" button opens the right sidebar (Fig. 4e) where, by clicking the "Add point" button it is possible to select a specific point on the scene to which connect an annotation. The annotation is saved with the name of the user that created the comment and the written text. All comments are stored in the Comments section to allow communication between different users, such as surgeons and engineers.

Add screws. By using the button with the screw icon the user can add screws to the scene to plan their insertion during surgery. Each created screw has connected tools to control the position and the rotation of the object (Fig. 4g). Moreover, it is possible to control the height and diameter of the screw.

Fig. 5 shows the planning of a case involving the pelvic anatomical region.

4. Impact

The idea behind the development of the *Precise* application was to provide a tool for the planning of cutting paths that could be used directly by hospital staff without the presence of an experienced CAD modeler. Using the *Precise* application the surgeon can define significant information for the creation PSI directly on the anatomical model specifically define in the initial design phase some important design constraints. Furthermore, the application was intended to enable the exchange of information between surgeons and engineers. The application has a user-friendly and intuitive interface and allows each user to manage multiple projects, considering that a surgeon can study multiple cases at the same time and an engineer makes surgical guides for multiple surgeries. In fact, the application can be easily used directly by physicians who, using the "Paint" and "Comments" functions, can communicate important information to be taken into account in the surgical guide modeling process; these communication functions can also be used in a post-operative verification phase. In addition, the program offers the possibility of measuring anatomical areas and inserting screws to indicate their correct position to the CAD modeler. To test the actual usability of the software, important metrics were tested, namely effectiveness, efficiency and satisfaction. The results showed on average a high degree of satisfaction and ease of use, proving that the software is a resource that can be easily exploited by medical staff. From the results shown above it is possible to notice that out of five users, one user failed to finish one of the tasks resulting in lowered satisfaction.

Compared with other 3D web modeling and visualization solutions, *Precise* differs in several aspects. Compared to some commonly used applications, *Precise* operates completely in the browser, therefore it is possible to work with it wherever the device can be reached by an internet connection. Another important aspect is that the user can locate and use all the features very easily and without having to learn complex programs intended for other types of use. Another aspect to consider is the possibility of expanding the functionality of the application by inserting new objects to manage the scene and the possibility of modifying the application so that it can be used to study and process different cases. In fact, the application was developed in such a way that it can be easily updated and reused. The program does not require the input of sensitive patient data, only an identification code and the uploading of the STL file of the anatomy. To ensure the actual use of the software within a hospital setting, it will be necessary to ensure a secure location of the data and an authentication system that complies with current hospital standards.

5. Conclusions

In this paper, a new application for navigating and manipulating three-dimensional models was presented with the aim of assisting preoperative planning for tumour resection in a collaborative framework. A full-stack cross-platform application usable on both computers and tablets was created. The application allows the physician to interact with 3D models in complete autonomy, making easily accessible 3D manipulation tasks that in other programs are complex and require a dedicated training phase. Some of the most important and interesting future developments include: the ability for two users to access a project at the same time, including the ability to view changes in real time when two users are connected to the same project, further simplifying the simultaneous design process; the integration of features to perform post-operational case analysis.

Funding

This work was founded under the project PRECISE (https://www.preciseproject.it/): co-founded by Tuscany Region, Italy under the call "BANDO RICERCA SALUTE 2018".

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Angela Busato reports financial support was provided by Tuscany Region.

Data availability

No data was used for the research described in the article.

Acknowledgment

The authors wish to acknowledge the Tuscany Region (Italy) for co-funding the PRECISE project (BANDO RICERCA SALUTE 2018), which originated and made possible this research.

References

[1] Javaid M, Haleem A. Additive manufacturing applications in medical cases: A literature based review. Alex J Med 2018;54(4):411–22. http://dx.doi.org/10.1016/j.ajme.2017.09.003.
[2] Volpe Y, Furferi R, Governi L, Uccheddu F, Carfagni M, Mussa F, Scagnet M, Genitori L. Surgery of complex craniofacial defects: A single-step AM-based methodology. Comput Methods Programs Biomed 2018;165:225–33. http://dx.doi.org/10.1016/j.cmpb.2018.09.002.
[3] Materialise Mimics, https://www.materialise.com/, [Online; Accessed December 2022].
[4] Sectra, https://sectra.com/, [Online; Accessed December 2022].
[5] mediCAD, https://medicad.eu/en/, [Online; Accessed December 2022].
[6] PeekMed, https://www.peekmed.com/, [Online; Accessed December 2022].
[7] Brainlab, https://www.brainlab.com/, [Online; Accessed December 2022].
[8] Blender, https://www.blender.org/, [Online; Accessed December 2022].
[9] ZBrush, https://pixologic.com/, [Online; Accessed December 2022].
[10] Sculpteo, https://www.sculpteo.com/en/glossary/sculptris-definition/, [Online; Accessed December 2022].
[11] Forger, https://www.maxon.net/en/forger, [Online; Accessed December 2022].
[12] ArmorPaint, https://armorpaint.org/, [Online; Accessed December 2022].
[13] 3DCoat, https://3dcoat.com/, [Online; Accessed December 2022].
[14] Polybrush, https://unity.com/features/polybrush, [Online; Accessed December 2022].
[15] Sherif T, Kassis N, Rousseau MÉ, Adalat R, Evans AC. BrainBrowser: distributed, web-based neurological data visualization. Front Neuroinform 2015;8(89). http://dx.doi.org/10.3389/fninf.2014.00089.
[16] Mecuris, https://www.mecuris.com/, [Online; Accessed December 2022].
[17] WebGL, https://get.webgl.org/, [Online; Accessed December 2022].
[18] Three.js, https://threejs.org/, [Online; Accessed December 2022].
[19] React, https://reactjs.org/, [Online; Accessed December 2022].
[20] Node.js, https://nodejs.org/en/, [Online; Accessed December 2022].
[21] PostgreSQL, https://www.postgresql.org/, [Online; Accessed December 2022].
[22] three-mesh-bvh, https://github.com/gkjohnson/three-mesh-bvh, [Online; Accessed December 2022].
[23] Raycaster Measurements, https://sbcode.net/threejs/measurements/, [Online; Accessed December 2022].
[24] Abran A, Khelifi A, Suryn W, Seffah A. Usability meanings and interpretations in ISO standards. Softw Qual J 2003;325–38. http://dx.doi.org/10.1023/A:1025869312943.
[25] Brooke J. SUS-a quick and dirty usability scale. Usab Eval Indust 1996;189(194):4–7.
[26] Sauro J, Lewis JR. Quantifying the user experience: Practical statistics for user research. Morgan Kaufmann; 2016, http://dx.doi.org/10.1016/C2010-0-65192-3.
[27] Nielsen J, Landauer TK. A mathematical model of the finding of usability problems. In: Conference on Human Factors in Computing Systems - Proceedings. ACM; 1993, p. 206–13. http://dx.doi.org/10.1145/169059.169166.
[28] Sauro J. 10 things to know about the single ease question (SEQ). Online; AccessedDecember2022.