GORAN DAMNJANOVIĆ

INSIDE SIMBION SUITE

Introduction — From the Player to the System

A basketball player does not train to become better at a test. The player does not step onto the court to prove that they are fast over five metres, strong in the squat, or explosive in a jump. The player steps onto the court to make the first move before the opponent, remain stable through contact, close space on defence, repeat a high-quality action when tired, and be ready when the game demands an answer.

That is why physical preparation has never been only physical preparation. It is the foundation for a basketball task. If it does not return to the game, it remains only a collection of exercises, tests and numbers.

FPK I established the methodological foundation: the player, the path from point A to point B, training, recovery, periodisation and the relationship to basketball. FPK II made the next move: a functional finding must receive a court translation, an operational model, a working card and a decision by the coaching staff.

SIMBION continues that path at the level of one player. Its role is not to create a generic programme simply because the user asks for exercises. Its role is to connect methodology, the player profile, diagnostics, correlation, position, context and the training base into training that has a reason.

But basketball does not end with one player and one training session. Behind every session stand the team, schedule, games, planning, technique, tactics, physical preparation, wellness, recovery, injury or risk, sleep, nutrition, mental state, equipment, travel, documents, budget and decisions by the staff. When all of this information is scattered across notes, messages, tables and separate tools, no decision has the full picture.

That is where SIMBION OS begins.

SIMBION OS is not another application, dashboard or collection of separate modules. It is a true operating system for sports work. The coach, staff and club enter one working environment, open what they need, use shared data, move between functions and create a confirmed output without leaving the system.

This book therefore has two parts. The first shows how SIMBION leads from the player profile to personalised training for the physical preparation of a basketball player. The second shows how SIMBION OS connects the player, team, basketball work, wellness, administration, knowledge, visual content and the final decision.

The application layer helps information become connected, explained and adapted when necessary. Methodology is the core, the training base is the infrastructure, the correlation model is the knowledge layer and the coach remains the decision authority.

> **You are getting methodology, a working structure and a decision with a reason.**

A catalogue that does not move anything

After a few seasons, every coach owns an archive whether they meant to build one or not. A phone full of clips from last Tuesday's session, a laptop folder from two clubs ago, a shared drive someone else set up and half-abandoned, a WhatsApp thread with a drill a colleague sent once and that was never found again. None of it is lost, exactly. It is simply scattered across enough places that finding a specific exercise costs more time than re-shooting it from scratch — so it gets re-shot, and the archive grows without ever becoming useful.

That is the problem Library is built against, and it is worth being specific about what it does and does not do before going further, because the two questions a coach asks first are usually the same: will this touch my files, and will it lock me into anything. The answer to both is the one guarantee the rest of this chapter keeps coming back to. Library is a local catalogue, not a storage service. It runs on the coach's own machine, reads a video, an image, or a document exactly where it already sits, and keeps a reference to it — a name, a set of tags, a place in the body, a note. It never copies the original, never renames it, never moves it, and needs no connection to anything outside that machine to do its job. Delete a catalogue entry and the file on disk is untouched; import it again whenever it is needed.

What changes, then, is not where the material lives but what a coach can do with it once it is catalogued. A warm-up filmed two years ago for one team becomes findable for a completely different one, not because it was moved anywhere, but because it was described once and can now be searched from any angle — by name, by tag, by the muscle it works, by a note written about it at the time. The archive stops being something a coach has to remember and starts being something the application remembers for them.

A tag, not a folder

A folder forces a single choice: this exercise lives here, and nowhere else. A warm-up drill used both in a strength introduction and a speed introduction would normally need a duplicate copy, and the two copies drift apart the first time one of them is edited. Library keeps the exercise as one entry and lets it carry several tags at once — hip, warm-up, no equipment — found by any of those words, in any combination, without ever having to decide in advance which folder it belongs to.

Tagging is built to work on many items at once rather than one at a time: mark a batch of cards in the Catalogue, type the tags once into the bar that appears at the top, and confirm. Existing tags on those items are not replaced — the new ones are added alongside them.

Every card also carries a set of muscles the exercise engages. That is not bookkeeping — an anatomical illustration is drawn directly from those muscles, and a player reading a real anatomical picture instead of a generic stick figure recognises the movement faster and performs it more accurately. Muscles are searchable in the same field as tags: typing quadriceps returns every exercise that engages it, regardless of what the exercise happens to be called.

A saved video is rarely used whole. The Review and Clips screen opens the source in a real player with two draggable handles under it; the video follows the handles as they move, so trimming is watched rather than typed. A first cut is provisional; only once the coach confirms the name and boundaries does it become a permanent clip, capped at five minutes and stored as its own new file — the source recording is never touched.

> **The exercise does not have a place. The place is chosen at the moment the training is built.**

Basket, block, and the basket that empties on its own

Searching turns up things that might be useful without forcing an immediate decision about them. Add to basket sets an item aside; the basket follows across every screen and empties itself when the application closes, because it is a desk to work on today, not a place to store anything permanently. What is worth keeping is saved as a block — a named, reusable set that a single click drops whole into a training, twenty exercises at a time instead of one.

A block does not copy the exercises it contains, only points at them. Renaming an exercise or adding a tag to it is visible in every block that references it, because there is still only one exercise underneath.

Origin: whose material is it

A club archive is rarely made up only of a coach's own recordings. A clip filmed by a colleague, a diagram copied from a federation manual, a video shared by another club — all of it is useful to have on hand, and none of it should quietly end up in a document sent outside the building. Every item entering the catalogue is marked with where it came from: a coach's own recording, which travels everywhere including into export; someone else's material, which works inside the application but never leaves through export; a link, which keeps only the address and opens at the source; or unmarked, treated as not yet decided and therefore withheld from export by default rather than assumed to be safe. When a catalogue is exported, anything not marked as the coach's own is listed at the bottom under what did not go in, with the reason — nothing is left out silently.

Nothing that is deleted disappears immediately, either. Material, a training, or a block removed from view lands in a recycle bin, restorable with one click. Only Delete forever removes it for good, and that is the one place in the whole application that asks for the item's name before it acts.

None of this depends on owning any other part of the Suite. Library stands on its own as an archive for a coach who simply has too much material and nowhere trustworthy to put it — searchable from the first day, safe with originals by design, and ready to be reused for as many seasons and as many players as the archive itself grows to hold.

The workshop, not the whiteboard

A tactic explained once, out loud, in front of a whiteboard, survives about as long as the marker takes to dry. Nobody photographs it, nobody can play it back before the next session, and the version an assistant remembers rarely matches the one the head coach actually meant. Video helps, but raw footage answers what happened, not what should happen next — and a still frame, however carefully drawn on, freezes one instant of a play that was never one instant to begin with.

Studio exists for the step in between: turning an idea about movement into something that can be watched, corrected, saved, and handed to someone else exactly as intended. Before the detail, the two questions worth answering directly are the same ones any coach would ask of a new tool. It runs locally and needs no connection to anything outside the machine it is installed on. And cutting a video inside it never touches the original recording — every trim, join, or telestration is saved as a new, separate file, so the source clip is exactly as safe here as it is in Library.

Everything Studio produces sits in four places, switched with a single click rather than four separate screens: the Board, where actions are drawn and animated; Video, where footage is trimmed, joined, and marked up; Archive, where notes and finished reports live; and Training, where all of it is written into an actual session.

A play is a film, not a photograph

The Board does not ask a coach to draw one finished diagram. It builds an action the way it actually happens — one moment at a time. Players and the ball are placed for the first moment; an arrow is drawn from whichever player moves, passes, or drives; clicking Next phase automatically carries every player who had an arrow to the end of it, and a fresh, empty phase is ready for whatever happens next. Repeating that as many times as the play requires and pressing Play turns the sequence into a real animation — the same phases, every time, shown to a player exactly the way they were drawn, not reconstructed from memory in front of a whiteboard a week later.

> **Not one static picture. A film with a few frames, played back the same way every time.**

A row of four balls switches the sport the Board is drawing for — basketball, football, handball, volleyball — and with it the court, the available surfaces, and the vocabulary of moves on offer. A cut looks different from a curl, a screen from a handoff, a cross from a through ball; each sport keeps its own set rather than forcing one generic vocabulary onto all four. Every move has its own fixed line style, too — a full arrow for movement, an open chevron for a pass, a dashed zig-zag for a dribble — so that opening a play a year later, or one an assistant drew, still reads correctly without asking what each line meant.

A saved play is filed in the Playbook under the categories that sport actually uses — attack, corner, seven-metre and the rest change with the sport selected — and can be reopened, edited, or pulled straight into a training from there. A ruler tool gives a real distance in metres for any line drawn on the court, useful whether the question is a tactical spacing or a conditioning distance. Extra courts can be added side by side, up to five half-courts or two full courts, for a transition or a two-direction drill, and the whole view shrinks automatically to keep everything on screen. Once a play is finished, Export video renders the same animation to a file that drops straight into the Video workspace like any other recording.

What a diagram cannot say

A drawing shows what happens. It does not, on its own, say why — against which opponent, in what context, what the trigger is, or how a coach will know it worked. Studio's Solution field sits directly under a training's header for exactly that reason: the goal, the roles, what a player should read and how they should respond, and how the idea carries over into the actual practice. Adding it to a training drops the whole plan — goal, roles, and the linked drawing — into that session's blocks without retyping any of it.

The same header carries a Cycle field, exactly as it does elsewhere in the Suite: name it once, and the 1 · 3 · 7 · 14 · 21 buttons create that many empty, sequentially dated trainings under the same name, opened and filled one at a time rather than built from nothing.

Video work is grouped by what it touches rather than left as a loose row of buttons: a single frame is marked up in Telestration; a whole clip is trimmed, joined, or turned into a slideshow; and a finished output — a composed training video or a version cut for posting — comes out the other end. Anything exported can be saved straight back into Studio's own clip list with one click, ready to attach to a training as proof of the work rather than left stranded on the desktop.

Archive is where the parts that never fit inside a drawing get written down — a notebook that saves itself while typing, and an export that produces a real PDF carrying every phase of every action alongside the written situations, not a system print dialog standing in for one.

Studio keeps its own storage, entirely separate from Library or Training Tools — nothing here is shared automatically with the rest of the Suite, by the same design that keeps every application answerable for its own material. It also runs on a phone with the same content behind it, the toolbar collapsing to the bottom of the screen and the board expanding to fill it, so a play sketched courtside is the same play open on the laptop back in the office.

TRAINING TOOLS

One basketball workspace, not three separate apps

Library catalogues material for any sport. Studio draws and animates plays for any sport. Both are built wide on purpose, because a club rarely runs only one team in only one discipline. But a basketball coach preparing a single session does not want to think about four sports and three applications — they want to open one worksheet, pull in a clip, draw a set, write the session, and be done. Training Tools is that answer: a basketball-only workspace that already contains a catalogue and a board inside it, rather than a fourth application that has to be stitched to the other two by hand.

It is self-contained by design. Nothing here depends on Library or Studio being installed at all — the catalogue, the drawing board, and the archive all live inside Training Tools' own storage, separate from either of them, so the answer to whether it works on its own is simply yes. It also runs without an internet connection, the built-in chat included, so nothing about preparing a session depends on a signal at the gym.

The home screen keeps that promise visible from the first click: the last worksheet opened, or a button to start a new one, sits above a list of everything prepared before it, with four tiles underneath — Exercises, Material, Studio, Chat. All four open the exact same workspace; they only choose which tab is active on the way in, because switching between exercises, material, the court, and the finished training is meant to be one click inside one screen, not four different places to remember.

Catalogue and court, without leaving the session

Adding material is a single step — pick a video or image from the computer and it uploads once, after which it stays available to every worksheet built afterward. The catalogue keeps recordings and images separate from clips, which are the trimmed, derived pieces made inside the Court tab's video tools rather than raw uploads; a search field at the top narrows either list as soon as typing starts.

The Court tab carries the same drawing logic as standalone Studio, because it is, in effect, Studio built into this workspace: one phase is one moment of a play, an arrow shows the movement out of it, Next phase automatically carries the players forward, and Play animates the whole sequence smoothly, phase into phase. The same line legend applies — a solid triangle for movement, an open chevron for a pass, a dashed zig-zag for a dribble, a crossbar for a screen — so a diagram drawn last season still reads correctly today. A vertical/horizontal toggle lets the same half-court or full court be viewed either way, useful when a session is being prepared for a phone held upright rather than a laptop screen.

> **Exercises, Material, Court, Video, Archive, Training — one row of tabs, one click apart, not four separate places to remember.**

The worksheet is the session

A worksheet starts by choosing a form — basketball, physical preparation, or combined — which decides what blocks it opens with; a basketball form gives an opening, main, and closing part, physical preparation gives its own working blocks, and combined leaves the structure open. From there, material is added only where it genuinely helps a block: a saved Court drawing, a clip from the catalogue, an image, alongside the coach's own text and notes, which carry equal weight to anything visual.

Saving keeps a working copy; Print or Save PDF produces a readable sheet; Export writes an actual backup file into storage, meant for sending on or keeping safe; and Import brings that file back as a new worksheet rather than overwriting whatever is currently open, so restoring an old session never risks the one being worked on right now. None of this asks a coach to think in file formats — the interface only ever talks about titles, material, plans, and sessions.

A Cycle field in the worksheet header works exactly as it does across the Suite: name it, then use the 1 · 3 · 7 · 14 · 21 buttons — or a custom number up to twenty-one — to create that many empty, sequentially dated worksheets under the same name, each opened and filled in turn. The saved list groups sessions of the same cycle together, ordered by date, and the calendar view shows the same days across a month.

What the app will not do

Originals are never copied, moved, renamed, or deleted automatically — a derived file only appears after a coach's own clear action. Training Tools does not send email, does not post to any network, and does not make a methodological decision in the coach's place; what to train and how to plan it stays entirely outside its job. The same boundary holds for its built-in chat, reachable either as a floating hexagon on every screen for a short answer, or as its own tile on the home screen for the full procedure: it answers how the application works — what to click, where something is saved — and nothing about what a session should actually contain.

Everything the app creates lands in one dedicated folder on the desktop, entirely apart from Library's or Studio's own storage: the working database, the catalogue, derived clips and board images, generated reports, and every exported worksheet, each in its own named place, none of it touched by hand.

PART I — SIMBION

1 Training with a Reason

A good training session does not begin with an exercise. It begins with the question of what the player needs to do better on the court.

Does the player need to get out of the stance faster? Hold position through contact? Close out a shot without arriving late? Keep the quality of the first step in the second half of a game? Does the player have strength but fail to transfer it effectively into a change of direction? Every one of those questions can lead to exercises. None of them should begin with exercises.

The physical preparation of a basketball player has value only when it returns to the game. Its result is not a beautiful programme on paper, but a player who is more prepared to execute a basketball task. That is why training quality is not determined by the number of exercises, the quantity of information or the complexity of a plan. It is determined by the relationship between the player, the player's state, the moment in the season and the work that genuinely helps the player progress.

Methodology is the way to keep that relationship from being lost. It does not allow a player to be reduced to one weakness. It does not allow a test result to become a label. It does not allow training to be assembled as a list of the best exercises we have ever seen. Instead, it establishes an order: first the player, then the finding, then the meaning of that finding, then priority, context, selection of work and verification that the work returns to basketball.

SIMBION is the application that turns this order into daily work. It does not begin with a request for exercises. It begins with the player and the player profile.

Personal data, sex, category, tests, position and coach observation enter the application. The relationship between abilities is then shown, critical gaps and training priorities are identified. Program determines what can be developed together and what requires caution. Context — number of days, sessions, games, season, microcycle, position, sex, category and basketball alternative — determines how much, when and in what form specific work makes sense.

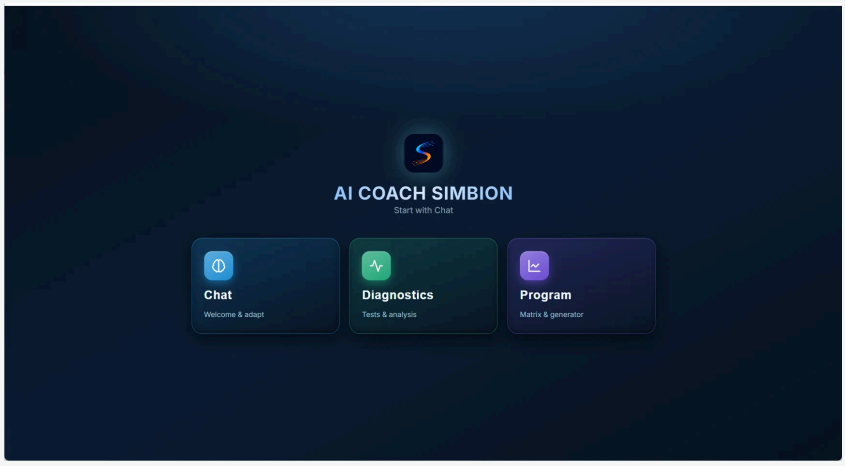
Only then does training appear.

The training database provides content. The correlation model protects methodological relationships. The decision engine connects data, rules and context

into a valid selection. The application then enables the user to see the result, understand the reason for selection and, when necessary, request a permitted training modification.

The coach does not lose a role in this process. On the contrary, the coach gains a better structure for what has always been the job: to see the player, understand the situation and make a decision.

> **Simbion does not begin with the question “which exercise should I give?”. It begins with the question “what does this player need now to become a better basketball player?”.**



SYSTEM ON SCREEN
Figure 1. Work does not begin with exercise selection. It begins by entering a system that first asks for the player.

When Training Quality Becomes Visible

Training quality is not visible only in whether a player finishes a session tired. It is visible in whether every part of the session is there for a reason.

Warm-up does not exist to fill time before the main work. It prepares what training will require later. Activation does not exist to look modern. It should set the body for the demand that is coming. The main section does not exist to collect as much content as possible, but to develop what is currently a priority without damaging the rest of the work. The final section does not erase fatigue, but it helps close the session according to the player's state and the obligations that follow.

That is why good training often looks simpler than the process that produced it.

Behind one exercise, several questions can stand. Is the player beginner, intermediate, advanced or elite in a particular factor? Is the deficit truly a priority or only a noticeable test result? Can that factor be connected to another factor on that day? Is the player in a phase that calls for activation, base work, a competitive microcycle or recovery? Does the player's position require more work

through a particular basketball pattern? Does the user want a classical athletic or basketball alternative?

When those questions are answered in order, training stops being improvisation.

That does not mean the application tries to create a perfect plan that never changes. Basketball work changes. The player responds, a game arrives, the schedule changes, equipment is not always available and the coach sees something new on the court. The system must therefore be stable enough to protect methodology and open enough to allow change when change is justified.

This is where the principle of base training and targeted additions becomes important. A player cannot train only what is not good every time. If the entire session became a punishment for a deficit, the development of the rest of the physical profile would be lost, work quality would decline and the player could easily stagnate. Base training protects continuity. Two or three targeted additions give direction to correction. They enter where they make the most methodological sense, without breaking the whole session.

In this way, the application does not promise a miracle. It enables consistent work.

> **Training has a reason.**
> The player should recognise the connection between the work done today and the task that must be solved better tomorrow.

2 The Player Does Not Start with an Exercise

The most common mistake in planning physical preparation begins before the first exercise is selected. A coach or player sees a weakness, finds several exercises and starts working immediately. Sometimes that can create a short-term feeling of work. But without a clear picture of the player, that approach has no direction.

SIMBION therefore does not begin with an exercise. It begins with the player.

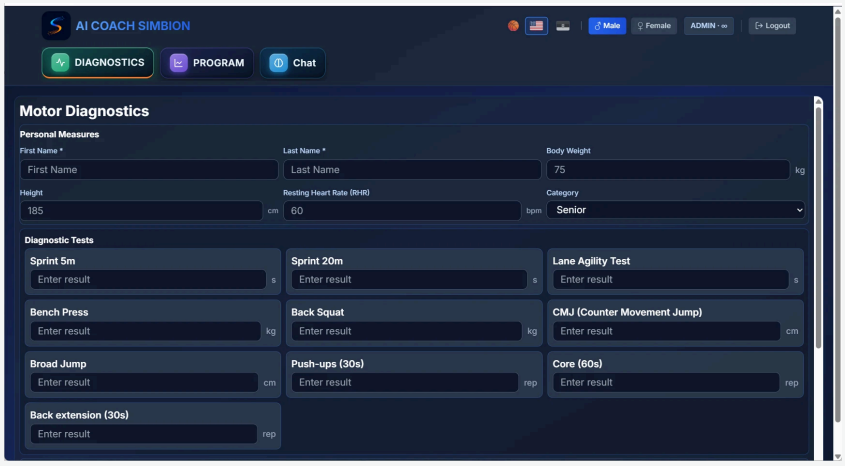
The first panel of the application is not an exercise catalogue. It is Diagnostics. This is where the initial working context is formed: personal data, body measures, sex, category, diagnostic tests, position and Coach Observation. None of those items is decoration in a form. Each adds one part of the picture that will later influence Program, correlation, context and training.

Personal data do not exist to turn the player into an administrative record. They establish the basis on which a result is read. Sex and category separate contexts in which the same result does not always mean the same thing. Body weight, height and morning heart rate provide an initial physical picture. They do not determine training on their own, but they prevent the player from being read as an abstract number.

Diagnostic tests introduce an objective foundation. In the Windows version, tests include the 5-metre sprint, 20-metre sprint, Lane Agility Test, Bench Press, Back Squat, Counter Movement Jump, Broad Jump, Push-ups in 30 seconds, Core in 60 seconds and Back Extension in 30 seconds. Every test carries information, but no test is the complete truth about the player.

The 5-metre sprint gives an image of the start and acceleration in the first steps. The 20-metre sprint shows the ability to accelerate through a longer distance, but it must not automatically be read as maximum speed. The Lane Agility Test records a pre-planned change of direction; it is not a direct test of reactive agility in open play. Jumps and strength tests describe particular force and explosive capacities, but cannot independently say how well the player transfers that force into a real basketball task.

That is why diagnostics do not end with the test. They move into analysis.



SYSTEM ON SCREEN
Figure 2. Personal data, body measures and tests create the starting foundation on which a result is read.

Position and Coach Observation

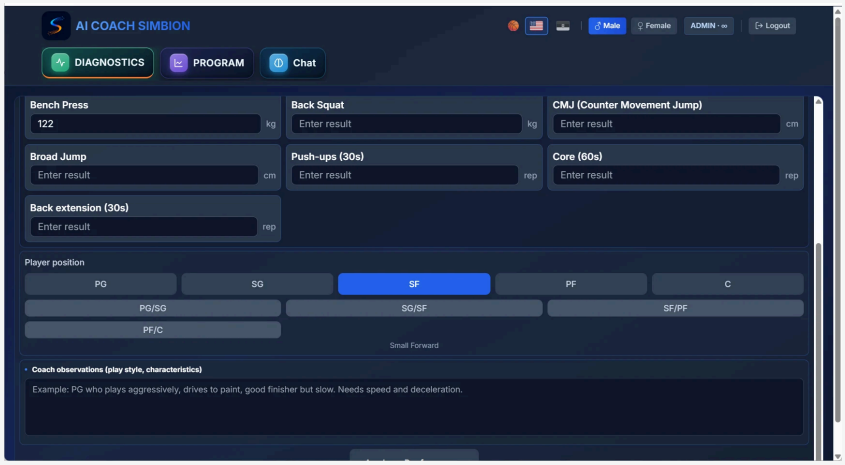
Player position is the next mandatory filter. The application allows an athlete to be selected as a primary or hybrid position: PG, SG, SF, PF, C, as well as combinations such as PG/SG, SG/SF, SF/PF or PF/C. Position is not entered to give the player a label. It remains active throughout the system. The same deficit does not carry an identical basketball meaning for a point guard, wing and centre. Acceleration, contact, change of direction, footwork, stability and capacity can be developed through similar physical principles, but their application on the court changes with the player's role.

Coach Observation is the panel in which the coach enters what is seen in real work: playing style, characteristics, a recurring situation, a problem the test does not register, or a task the player needs to solve better. It is not a privileged input and it does not override diagnostic results. It is an equal contextual panel.

The distinction matters. If diagnostics show a good sprint result, a general statement that the player is "slow" cannot cancel the finding. But the coach can specify that the player is late in a close-out, loses the first defensive step, reacts poorly to a signal, or establishes body position too late. That observation does not change the fact of the test. It explains where the transfer of ability should be checked in basketball play.

At this point, the application does not yet select training. It builds the profile from which training can be selected with reason.

> **A player is not a sum of tests. Tests, position and coach observation are different views of the same work problem.**



SYSTEM ON SCREEN

Figure 3. Position remains a permanent filter, while Coach Observation completes the player profile without cancelling a valid finding.

3 From Data to Feedback

Data alone do not train a player.

A sprint, jump or strength-test result can be precise, but the precision of a number does not automatically make the decision clear. It does not help a player to hear only that a result is below a certain norm. It does not help a coach to receive a list of numbers if the next work step cannot be seen from it.

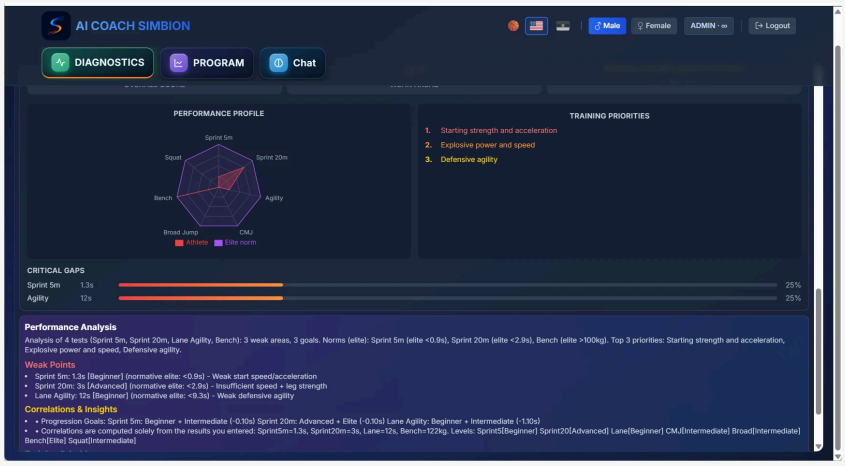
That is why, after Diagnostics, SIMBION shows the Performance Profile. The radar does not exist to create an attractive visual. Its function is to show, in one image, the relationship between what has been measured in the player and the reference level the application uses to read the result. The player should not see the radar as a verdict. The player should see it as an initial map.

The map does not only show where the player is weaker. It also shows what is already ready for work, where imbalance exists and which abilities should be connected for the player to take the next step.

On the Windows screen, the application shows Training Priorities and Critical Gaps alongside the radar. These are two different but connected outputs.

Critical Gaps identify measurable differences that need attention. If the 5-metre Sprint result is below the desired level, the application must not only say “work on speed”. It needs to indicate that the issue concerns the start and acceleration. If the Lane Agility Test is weaker, the system must not pretend it has measured the complete defensive reaction in open play. It can show that a pre-planned change of direction needs further attention, while the coach must connect that finding to the real demand of the game.

Training Priorities are the next step. They translate several results into an order of work. In the example shown, the priorities are starting strength and acceleration, explosive power and speed, and defensive agility. That output does not mean the player performs three completely separate programmes tomorrow. It means the system sees three areas that should be connected through methodologically valid training.



SYSTEM ON SCREEN
Figure 4. The radar shows the relationship between abilities; priorities give direction to work; coach and player determine how that direction returns to the game.

Three Levels of Feedback

Feedback has three levels. The first is the physical level. It says what has been measured and where a difference exists. The second is the training level. It explains what that difference changes in the choice of work. The third is the basketball level. It connects work with concrete situations: the first step, moving out of a defensive stance, stopping and accelerating again, closing out a shot, contact, jumping or the second effort in a sequence of actions.

The player should hear all three things, but not in the same language as the coach. The coach needs the precise finding and priority. The player needs a clear message.

> **Message to the Player**
> Your first step is not strong enough compared with the other abilities you already have. The next work will not be only sprinting. We will work on creating force faster, keeping it through a change of direction and transferring it into the first defensive step.

That type of feedback does not promise that a test will solve the game by itself. It gives direction and a reason for work. The radar therefore does not end the story. It is an entry into conversation, Program and training.

4 Program: Analysis Does Not Stay on the Screen

Analysis is useful only if it changes the next training session.

Radar, Critical Gaps and Training Priorities give direction, but they do not deliver the work on their own. This is why the process moves from Diagnostics into Program. It is not a second, disconnected form. It is the place where the analytical result is translated into the working language of physical preparation.

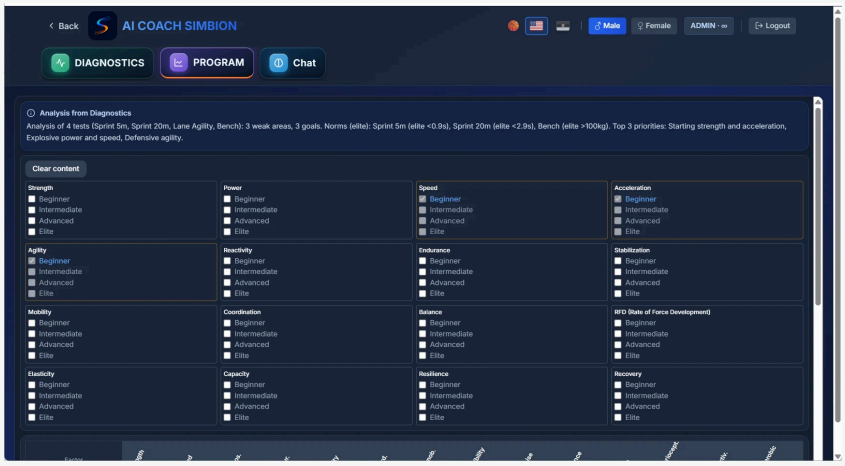
At the top of the Program panel, a short “Analysis from Diagnostics” record remains. This preserves the trail: the user can see which tests, levels and weaker areas produced the initial recommendation. Analysis is not copied as a final verdict. It is the starting foundation that Program organises further.

Program uses sixteen parameters across four levels: Beginner, Intermediate, Advanced and Elite. The parameters include Strength, Power, Speed, Acceleration, Agility, Reactivity, Endurance, Stabilization, Mobility, Coordination, Balance, Rate of Force Development, Elasticity, Capacity, Resilience and Recovery.

Their function is not to turn a player into a list of sixteen problems. They allow the player’s state to be read more broadly than through one isolated test.

Some parameters come directly from diagnostics. When a test has been entered, the application carries forward what has been objectively tested. The remaining parameters can be marked through expert judgement. This is not an invitation to arbitrariness. It is recognition that no diagnostic package measures everything that matters in work with a player.

When no test exists for a particular factor, expert input must not pretend to be a new test result. It marks a working level that the coach wants to include in Program, with the responsibility that the choice makes sense in relation to the player, position, observation, season and the rest of the data.



SYSTEM ON SCREEN
Figure 5. Program does not create a new diagnosis; it organises what has been tested and what expert context justifiably includes in the work.

Four Levels Give Work a Measure

Coach Observation remains one equal source of context here. It does not receive privilege over diagnostics, Program or the correlation model. Coach observation can explain why a particular parameter should be included in the work: for example, a player may have a good physical result but arrive late when setting body position in a defensive situation or lose balance in contact. The system does not erase the test result. Program simply receives a broader picture of what should be developed.

Beginner does not mean a bad player. It means that a particular factor is at a level from which work should be built carefully. Elite does not mean the factor no longer needs training. It means the player already has a high ability in that area, so training should maintain it, transfer it or connect it with another demand. Between those two ends are Intermediate and Advanced, the real working levels in which most players spend much of their development.

> **Diagnostics measure. Program organises what measurement and expert context mean for the next work.**

Program does not create a perfect profile for display. It creates a working profile for training selection. In the next step, those parameters do not remain isolated. The correlation model checks what can be developed together, where work needs to slow down and which two or three additions make sense inside existing training.

5 The Correlation Model and Context

A player does not train factors one by one as if none of them were related to another.

Starting strength affects the first step. The first step affects how quickly a player can create an advantage or close down an opponent. But a first step without deceleration control, balance and re-acceleration does not solve the whole task. Likewise, strength without transfer into movement, explosiveness without position control, or capacity without recovery can remain abilities that do not return well enough to the game.

That is why Program leads into the correlation model.

In the Windows version, the model is shown through a 15×15 correlation matrix. On Android, the same logic is shown as a list. Only the form is different. The workflow, methodology and selection rule remain the same.

The matrix does not tell the user what the “best exercise” is. It shows the relationship between factors. The ++, +, ±, – and 0 markings represent different levels of compatibility. ++ shows a strong methodological relationship; + a compatible combination; ± a relationship that requires caution and depends on context; – a combination that should not be forced into the same session; and 0 a neutral relationship.

The user does not need to memorise the entire matrix. Its value is precisely that methodological logic remains available while a concrete training session is being built.

Correlation does not lead to a programme overloaded with additional content. On the contrary. The rule of the application is to preserve base training and add no more than two or three targeted additions through correlation focus. Those additions are not the whole training session. They are a precise change inside a session that already has its own order, dose and purpose.

This protects player development from two extremes. The first is to ignore a weakness completely because it is easier to repeat familiar work. The second is to turn every session into work only on the deficit. In both cases, the player loses continuity. In the first case, the player does not progress where it is needed. In the second, the player loses what is already done well and can easily stagnate.

SYSTEM ON SCREEN
Figure 6. The matrix does not select training alone. It protects relationships between factors while context determines what makes sense today.

Context Provides the Moment

The correlation model therefore asks: what can help the priority now without damaging the rest of the work?

The answer does not depend only on factors. It also depends on context.

At the bottom of the Windows panel, the user enters number of days, total training sessions, games per week, season, microcycle, position, category, sex and, when needed, selects Basketball alternatives. These parameters are not a formality. The same deficit does not receive the same answer when the player is in an activation microcycle before a game, in a base period without competitive pressure, or in a week with several games.

If there are few days and more games, training must preserve freshness. If there is broader room for work, a larger dose can be built. If the user selects Basketball alternatives, the application does not change the objective of the work; it changes the form through which that objective is developed. Instead of a classical athletic form, certain content can take a basketball-appropriate form.

Position remains an active filter here as well. For an SF, as in the example shown, the first step, strength through contact, defensive angle and change of direction carry a different pattern of demands than for a PG or C. The physical principles remain, but their basketball application gains measure through the player's role.

> **Correlation provides the connection. Context provides the moment. Only together do they give training selection its meaning.**

6 A Decision That Can Be Explained and Adapted

Training is not finished at the moment it appears on screen.

Only then does the user receive what must be clear: what is being done, why it is being done and how the work can be adapted if the real day changes.

Delivered training has structure. It is not only a long list of tasks. It shows an introductory section, preparation for the main demand, the main work and a closing section. When correlation additions enter the programme, they must not hide the basic purpose of the session. Base training remains the carrier of order. No more than two or three targeted changes move the work toward the priority that matters most on that day.

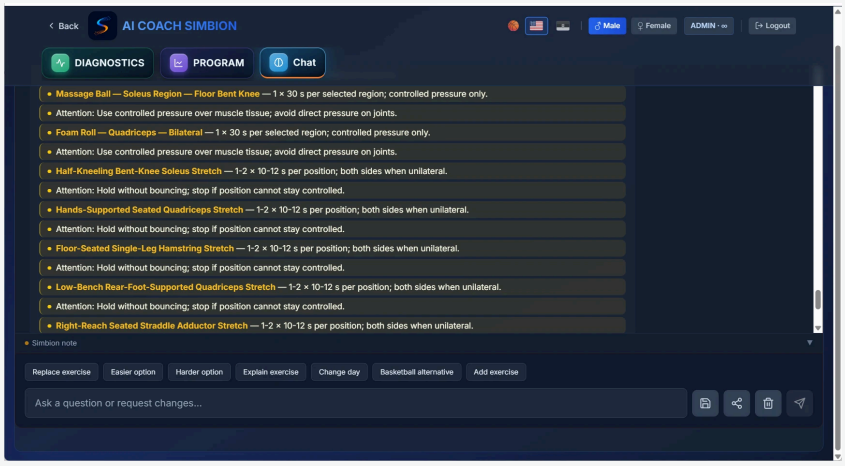
That is the answer to the question “Why this?”.

Why this training for this player now? Because the player profile showed a particular relationship between abilities. Because Program determined working levels. Because the correlation model connected factors without damaging the rest of the session. Because context limited or opened the dose. Because position gives physical work basketball meaning.

This explanation needs to exist for both coach and player.

The coach needs to see the professional logic of selection: which priority is being addressed, what remains base work, which two or three additions have been made, what is intentionally not being developed that day and which response should be monitored. The player needs a shorter, clearer message: what is being developed today and where it should help in the game.

For example, if the focus is starting strength and acceleration together with defensive agility, the player does not receive the message “today you work on speed”. The player receives the message that today’s work develops the ability to create force faster in the first step, control position while decelerating and accelerate again when closing down an opponent. That kind of message returns training to basketball.



SYSTEM ON SCREEN
Figure 7. Chat does not invent new training. It protects the objective of already delivered work while the user requests a permitted modification.

Chat Is the Continuation of a Decision

Chat comes after training has been delivered. It is not the first place where a plan is built from a blank page. Its function is to allow the user to make permitted changes when real working conditions require adaptation.

The screen makes clear options available: Replace exercise, Easier option, Harder option, Explain exercise, Change day, Basketball alternative and Add exercise. There is also space for a specific user request.

Every one of those changes must protect the objective of the work. If the user asks for an easier option, the easier version must not change the reason the exercise is in the training session. If the user asks to replace an exercise, the replacement must remain inside the same methodological task. If the user asks to change the day, the dose and order of work must make sense again in relation to the schedule. If a basketball alternative is selected, the objective remains the same while the form of work is translated into a more basketball-specific expression.

That is why Chat is not generic conversation. It is the continuation of an already made decision.

> **A good modification does not break training. It protects the reason for training while adapting it to the real day.**

This is the final value of the application. The user does not only receive a plan produced by the system. The user receives work that can be understood, explained to a player and adapted without losing methodology.

After that comes monitoring through the next work: new observations, new tests, a new game, a new player response and a new decision. The system does not close on one session. But every next decision must begin in the same order — from the player, the player's context and the reason for work.

Part I Conclusion

SIMBION is not an exercise generator that begins with an empty request. It is an application for the physical preparation of basketball players that turns methodology into a daily order of work.

The path is clear: player, diagnostics, position, Coach Observation, performance analysis, Program, correlation, context, training selection, permitted modification and monitoring.

On that path, methodology remains the core. The training database provides content. The correlation model protects relationships. The application layer connects, explains and adapts. The coach remains the decision authority.

When a player receives training created in this order, the player does not receive only a task for today. The player receives work that explains the relationship between the player's state, the basketball demand and the next step.

> **From Player to Training**
> Training is not the end of the process. It is the next concrete decision in work that continually returns to the player through results, observations and context.

SIMBION BASKETBALL COACHING

The tactical half of the job

Part I of this book follows a player through physical preparation — diagnostics, a profile, a programme, a correlation model behind it, a chat that explains why the application chose what it chose. That answers one half of a coach's real question. It says how ready a player's body is for the work. It does not say what that player should actually run on the court, how a new move gets taught until it holds up under pressure, or which drill builds the piece a set keeps failing on. That second half — tactics, technique, and the drills that connect them — is what Simbion Basketball Coaching is built to hold.

It is not a separate application bolted onto the side of Training Tools. It inherits the same base in full — the same catalogue, the same drawing board with phases and animation, the same worksheet — and adds three basketball-specific columns on top of it: Tactics, Technique, and Drills. A coach who already knows Training Tools is not learning a new tool here, only a wider one.

Tactics, technique, and the drills between them

Tactics is where a set or a defensive situation is worked out — the read, the trigger, the spacing, what changes when the opponent adjusts — using the same Court board and phase-by-phase animation already familiar from Studio. Technique is the individual side of the same coin: how a specific skill is taught and corrected, from first cue to game-speed execution, independent of any particular set. Drills is what turns either one into something a team actually runs at practice — the repeatable piece of a session built to train exactly the tactical read or the technical detail identified in the other two columns, rather than a generic exercise picked because it was available.

A basketball-specific assistant sits alongside all three, the same way the coaching chat does elsewhere in the Suite — reachable for a quick answer or a longer conversation, carrying basketball context with it, but never given raw access to a coach's media. It can be asked how something in the workspace works or discuss a tactical idea in the abstract; it never receives a file path or a media byte directly, and every change it might suggest still has to be confirmed by the coach before it becomes part of a real session.

> **Part I answers how ready the player's body is. Simbion Basketball Coaching answers what that player should actually do with it.**

One final training, not two

The risk with any add-on column is that it quietly produces its own separate output, so a coach ends up assembling a session from two unrelated exports instead of one. Simbion Basketball Coaching does not do that. Whatever is built in Tactics, Technique, or Drills — like whatever is built in Studio — lands in the same final Training workspace already used across the Suite. A coach can write in additional notes there, print it, export it, and share it exactly as with any other training; there is no shortened or parallel version of that final step reserved for the tactical side of the work.

Library's rules travel with it unchanged: original media is never copied or altered automatically, and a derived file only ever appears after the coach's own confirmed action. What changes is scope, not principle — Tactics, Technique, and Drills give the tactical half of the job the same structure, the same board, and the same safeguards that Part I already gave the physical half.

1 Basketball Has Become Too Complex for Separate Tools

Basketball has never been only training.

Behind one training session stands a player. Behind the player stands a team. Behind the team stand the schedule, games, work plan, technique, tactics, physical preparation, recovery, injury or risk, sleep, nutrition, mental state, equipment, travel, documents, budget and decisions that are often made under time pressure.

The problem is not that the coach, staff or club have no information. The problem is that information is scattered. One item is in a note, another in a table, a third in a message, a fourth in an old programme, a fifth in the medical staff's memory, a sixth in the travel schedule and a seventh in a training video. When work is managed through disconnected tools, no decision has the whole picture.

That is not a technical problem. It is a work problem.

FPK I established the methodology: the player, the path, training, recovery, periodisation and the basketball task. FPK II required that a functional finding receive an operational translation into action on the court. SIMBION organised that methodology for the individual physical preparation of a basketball player.

SIMBION OS begins when solving one training session or one player is no longer enough.

It does not arrive to replace the coach, staff or club. It arrives to connect their work. It enables the player, team, diagnostics, training, wellness, administration, content and final decision to remain in the same working environment.

> **SIMBION OS is a true operating system for sports work.**

It is not a dashboard that only displays data. It is not a collection of separate applications. It is not a catalogue of documents. It is a working environment in which the user enters with a real task, opens what is needed, uses shared context, moves between functions and creates a concrete output without leaving the system.

When a coach enters the Player column, there is no need to remember which other application contains the plan. When a medical staff member checks status, there is no need to lose the connection with training. When it is necessary to draw the court, prepare video, open a drill, make a schedule, check equipment

or send a final document, the system does not force the user to start again in another tool.

That is the difference between software that has many functions and an operating system that has a workflow.

SIMBION OS should therefore first be understood as a working surface. The user does not open another programme. The user enters a workspace. The home screen does not announce sixteen disconnected products. It opens sixteen workspaces that can be used independently, while remaining connected through the same context.

The next chapter shows how that space stays open while the user works: through the left working palette, the Simbion–RAG–Studio kernel and columns that are not destinations but tools available when they are needed.

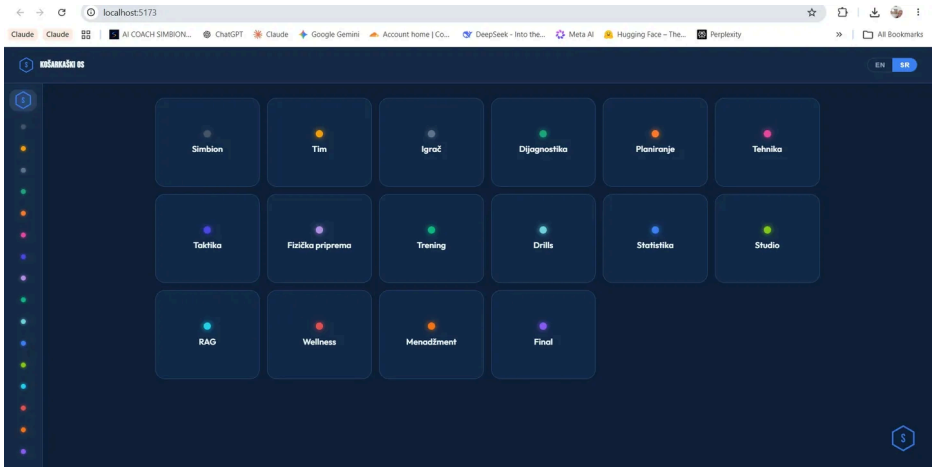


Figure 1. The home workspace: 16 connected working environments inside one system.

2 The Working Palette: Columns Are Not Destinations

If SIMBION OS is a working environment, its columns are not separate rooms that the user enters and finds difficult to leave. They are tools available inside the same workspace.

That matters because the real day of a coach, staff or club never follows a straight line. A coach may begin with a player and immediately need the latest training session. Training can raise a question about the schedule. The schedule can change the planned load. A change in load can open wellness, medical status or nutrition. The same situation may create a need for a court diagram, video, working document or confirmed output for the staff.

In disconnected tools, every one of those changes interrupts the workflow. The user closes one programme, finds another, re-enters context, transfers a file, checks whether the version is correct and tries not to lose the connection with the original question. Time is not lost only through clicks. The picture behind the decision is lost.

SIMBION OS addresses that problem through the working palette.

When the user enters one column — for example Player, Team, Diagnostics, Planning or Training — the left working palette remains available. It opens as a floating panel from which the user calls the next tool when it is needed. There is no need to leave the current work in order to find another part of the system. There is no need to remember where content is physically stored. There is no need to move into an entirely different environment simply because a decision requires another type of information.

> **Columns are not destinations. They are tools in the same workspace.**

That sentence changes the way the system's home grid is read. Sixteen tiles do not represent sixteen disconnected products. They represent sixteen workspaces with their own clear functions, while sharing context, knowledge and the ability to move into another part of the system.

For example, the Player column does not exist only to store a profile. It is where identity, history, characteristics and individual context become visible. But as soon as the coach needs the player's relationship with the team, Team opens. When a finding or status is needed, Diagnostics opens. When training content

is required, the user moves into Planning or Training. When the plan's real feasibility must be checked, wellness or Management enters the picture. Everything remains part of the same working story.

The same principle applies to the team. Team is not only a roster on screen. It is a starting point for role allocation, work planning, communication, statistics, scheduling and the real organisation of the day. From Team, the user does not move "into a new application". The user opens the next tool the decision needs.

The working palette therefore has no decorative role. It is the operating mechanism that protects continuity of work.

The User Does Not Travel Through the System. The System Follows the Work.

In classical navigation, the user follows the software structure. The user needs to understand where each function lives, in which order it opens and how to return. In an operating system, the logic is reversed. The system follows the user's task.

If the task is to prepare training, the user may begin in Planning. If diagnostic context becomes necessary during the work, it opens from the palette. If a court, diagram or clip needs to be created, Studio opens. If a final version must be confirmed, the work ends in Final. The initial task is not lost; it expands through the information and tools it actually needs.

This does not mean that every column performs every role. On the contrary. Each keeps a clear professional function. Diagnostics reads the state. Training organises work. Medical manages health context. Nutrition supports load. Management checks resources and organisation. Studio creates visual content. Final stores confirmed output.

The power of the system is not in removing these differences. Its power is in keeping the differences connected.

A coach therefore does not need to become a system administrator. The staff do not need to become an IT team. Everyone uses the function that belongs to their work while the shared environment protects the connection between all functions.

What the Working Palette Changes in a Real Day

Imagine a simple situation. A coach plans training for the day after a game. In Training, the coach opens the previous work and sees the basic structure of the next session. Medical is then called from the left palette to check the status of a player who had contact in the previous game. Nutrition opens to examine whether recovery and intake support the planned dose. Management is checked because the hall is available for less time than expected. From the same work-

space, Studio opens to adapt a court or visual task. When the decision takes its right form, the work ends in Final with material the staff can use.

None of those steps is a revolution on its own. Their sequence is the real job. When it happens inside one system, the decision remains readable, checkable and connected to the reason it was made.

The working palette does not accelerate work only because it shortens the path. It improves decision quality because it reduces the chance that an important part of the context remains in another file, another message or only in someone's memory.

Operational Sequence: From Training to Diagnostics Without Breaking Work

The coach is in the Training column, composing a session. The training form is open: who the work is for, the main focus, session duration and the arrangement of the introductory, main and closing parts of the work. While selecting content, the coach needs a piece of information from Diagnostics.

The coach does not leave Training and does not open another application. A click on the left sidebar opens a small floating Diagnostics panel next to the current work. Inside that panel, the coach can find the finding, profile, indicator or other information needed for the decision. The original Training context remains open beneath the panel.

Once the needed information has been checked, the coach closes the panel and continues in the same form. There is no return to the start, no search for another file and no repeated data entry. The diagnostic information serves the decision being made in Training.

The same principle applies to every column. Planning can open Team. Team can open Statistics. Training can open Wellness, Management or Studio. Each time, the user calls only the missing function, uses it in a small working panel and continues from the exact point at which work was interrupted.

> **The working palette does not interrupt the workflow. It brings the exact function needed at the moment of decision.**

System Boundaries Protect Responsibility

SIMBION OS can bring a data point, tool or existing content closer. It can connect workspaces and prepare an output. But it does not remove professional responsibility.

The coach still determines the priority. The medical professional still carries responsibility for the health context. The nutrition and mental-performance spe-

cialists still contribute the professional content of their disciplines. Management still decides on resources, obligations and organisation. The system protects their decisions from remaining isolated from one another.

That is the real purpose of the working palette: not to demonstrate how many options the system has, but to make the right option available at the moment a decision genuinely requires it.

The next chapter moves beneath the surface of that work. It explains the SIMBION OS kernel: Simbion, RAG and Studio — three elements that help knowledge, context and creation avoid becoming dispersed through disconnected tools.

3 The Kernel: Simbion, RAG and Studio

A working environment alone is not enough. For a system to truly connect the work of a player, team and club, it needs a shared layer that protects context, knowledge and the ability to turn information into usable content.

That layer is the SIMBION OS kernel: Simbion, RAG and Studio.

They are not three separate products. They work as three different functions of the same core. Simbion helps the user ask a question, understand a situation and initiate the next step. RAG holds the knowledge and materials from which the system draws context. Studio turns a working idea into a court, clip, diagram, biomechanical visual or another piece of content that can be used in real work.

Final stands at the end of that path. It is not a fourth kernel, but a confirmed output: the place where material arrives once it is ready to be stored, printed, sent or used further.

> **The kernel does not make a decision instead of the staff. It protects the context from which a better decision can be made.**

Simbion: A Present Assistant in Shared Context

Simbion is the system's master assistant. Its role is not to be an ordinary conversation window. It works with the context the user already has open inside SIMBION OS.

When a coach works with a player, Simbion can help locate a previous finding, explain the relationship between state and plan, identify the next step or prepare a working proposal. When staff work with a team, it can help locate relevant content, compare sources, organise a workflow or prepare material for discussion. When a club prepares an administrative output, it can help prevent the work from dispersing across separate documents.

What Simbion is not matters. It is not a replacement for a coach, doctor, nutritionist, psychologist, manager or any other professional. It does not turn a complex situation into one automatic verdict. Its value lies in accelerating access to context and helping the user formulate a more precise question.

If a coach asks to change training, a good system should not return a generic answer that could apply to any player. It should understand what is open, which context is already known and which boundaries must be protected. If Player is open, Simbion starts from the player. If Team is open, it starts from the team. If Training is open, it starts from the work that has already been composed.

In that way, Simbion follows the user's real job, not only the text entered into one field.

RAG: Knowledge That Remains Available to Work

RAG is the shared layer of knowledge and content. It is where documents, notes, working cards, materials, clips, explanations and other sources physically live for the system to use when needed.

Its role is not to create another library that no one opens. Its role is to prevent duplication, version loss and the search for the same material through different folders, messages and private notes.

Once content has a place in RAG, other columns use it through the needs of the work. Planning can take an existing working card. Training can take an exercise or previous plan. Studio can take content and turn it into visual material. Simbion can call the relevant part of the context while helping a user. Final can receive confirmed output with a clear trace of where it came from.

RAG is therefore not only storage. It is the system's shared memory.

Without shared memory, every new decision begins as if nothing had happened before. With shared memory, the user does not need to explain the basic context again every time a new tool is opened.

This especially changes staff work. Good work often does not fail because knowledge is absent, but because knowledge does not reach the person who needs it at the moment of decision. RAG brings relevant material closer to work instead of forcing work to search through the system for material.

Studio: Where Knowledge Becomes Visible Work

Studio is the workspace for creating and processing visual content. It can prepare a court, diagram, clip, video, biomechanical visual and other materials that help an idea be explained, trained or shared.

Studio is not an add-on intended only to make the system look modern. In basketball work, visual content is often part of the decision. A coach needs to show a court arrangement. A player needs to see a movement pattern. The staff need to compare a situation. A video or clip can explain a detail that a long message cannot.

When Studio is part of the same OS, visual content does not emerge outside context. It can begin with a player, team, existing training session, working card, diagnostic finding or content from RAG. The visual is then not only an attractive illustration. It becomes working material with a reason.

Imagine that a coach wants to prepare a short explanation of a defensive close-out for the players. In a conventional workflow, the coach needs to find or create

material in another tool and then return it to team communication. In SIMBION OS, the coach can begin from existing training or a tactical task, open Studio and create material that remains connected to the work from which it emerged.

This does not mean Studio removes the professional meaning of visual content. On the contrary. The coach still decides what needs to be shown, in what order and with what message. Studio shortens the path from a professional idea to material that can be used on court.

Operational Sequence: From Clip to Movement Correction

A coach can enter Studio from any column when a decision needs video. For example, while working in Training and checking the quality of a change of direction or exit from a stance, the coach opens Studio from the left working palette. The original Training context remains available; Studio opens as the workspace for material that needs to be analysed or explained.

The clip enters the workspace and only the part that matters for the task is selected. Instead of reviewing the entire recording, the coach can choose the start and end of the relevant action: first step, deceleration, change of direction, landing, shot, contact or another work segment.

Within the selected segment, the system follows the relevant player and movement. The view can place points, lines and angles over the body to make position, direction, segment relationships and the moment at which movement diverges from the required pattern visible. The coach does not receive a replacement for professional judgement; the coach receives clear visual evidence from which a correction can be explained.

The coach can then record the correction, prepare a frame or short clip, connect it with the player, training session or working card and send it to Final as material for individual work, staff discussion or communication with the player.

> **Studio does not create a clip for the sake of a clip. It turns movement into a visible task that the coach can explain and the player can understand.**

Final: Confirmed Output, Not a New Beginning

Final is where confirmed work arrives. It can be a training session, working card, schedule, document, visual material, report or another decision ready for use.

Its function is not to hide the path that led to it. On the contrary. Final prevents output from being only the last file in a chain of versions. It is confirmed material that the staff can find, use and return to when needed.

The distinction is substantial. When a final document is sent from a disconnected tool, the connection with the source, context and reason for change is often lost. When Final belongs to the same system, output remains part of the working story.

One Core, Multiple Professional Perspectives

Simbion, RAG and Studio do not require every user to work in the same way. A coach will often need a plan, training, court or video. The medical staff need status and documentation. Management need schedules, resources and confirmed documents. Each sees a different part of the work, but no one works outside the shared context.

That is the kernel's greatest value. The system does not try to erase differences between professional roles. It allows each professional role to contribute to the same decision without the club constantly transferring content from one tool to another.

The next chapter shows how that shared work is organised through the equal categories of SIMBION OS: player and team, basketball and training work, wellness, administration, the kernel and Final.

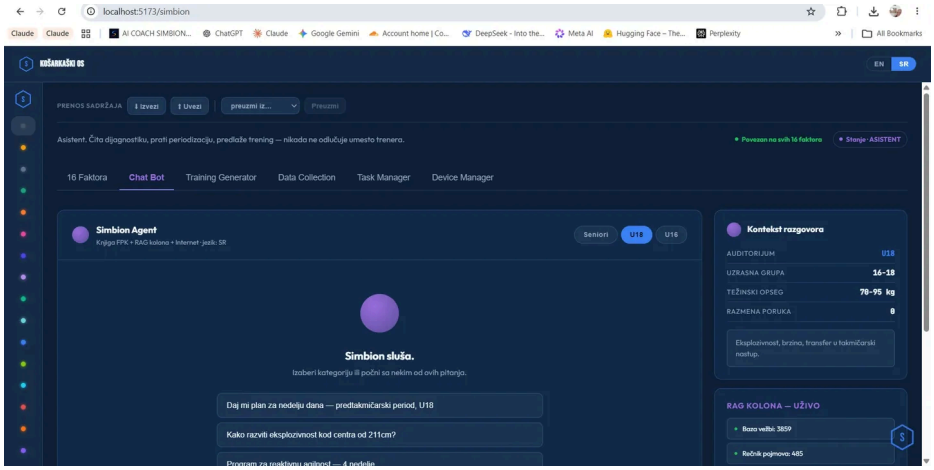


Figure 2. Simbion works alongside the context of the task and available system knowledge.

4 The System Is Broad Because Real Work Is Broad

SIMBION OS does not organise a club around one screen and one professional discipline. It organises the club around real work.

That is why the system has equal categories: player and team; basketball and training work; wellness; administration; the kernel and Final. None of them is an add-on opened only when time remains. Each affects what a club can do today, what it can plan for next week and how a decision can be followed through a season.

The connection between player, team, diagnostics and training is easiest to see. It is the natural language of sports work. But the real day of a club shows that a plan cannot be delivered if wellness does not allow the load, if administration has not secured space, equipment, transport or documentation, if the schedule does not leave time or if important content is unavailable when needed.

That is why the breadth of the system is not excess. It is the response to the breadth of the problems a club actually solves.

Player and Team: A Decision Begins with Those It Affects

The player is the individual context of work. This is where profile, history, state, development, position, obligations and response to previous work are read. The team is the shared context. This is where relationships among players, roles, load distribution, schedule, group and the real frame in which an individual plan must work become visible.

Diagnostics and Statistics do not exist to bury the system in numbers. Their function is to help recognise state, direction and priority. Data matter when they change the next decision. If they do not change a decision, they remain numbers without work.

The system therefore does not separate an individual profile from the team picture. A player may have a particular deficit or need, but the dose and moment of work gain meaning only when seen in relation to team obligations, schedule and the real course of the season.

Basketball and Training Work: Content Must Not Remain a Catalogue

Planning, Technique, Tactics, Physical Preparation, Drills and Training form the working chain from idea to court.

Planning gives time frame and order. Technique and Tactics protect basketball content. Physical Preparation develops capacity that returns to the game. Drills provide precise working units. Training turns them into a concrete session with order, dose, purpose and a place in the wider plan.

These columns are not a catalogue of what a club knows. They are the way knowledge becomes work.

A good exercise alone is not a system. A good training session alone is not a season. For work to have meaning, the exercise must know why it was selected, the training session must know where it belongs and the plan must know who it affects and under what conditions it is delivered.

SIMBION OS keeps that sequence from breaking. The user can begin from a plan, open a tactical demand, connect it with physical preparation, take a drill and compose training. Or the user can begin from training and return to the plan when checking whether a decision has the right place in the microcycle.

Wellness: State Determines Whether the Plan Can Be Delivered

Wellness consists of Medical, Mental and Nutrition. These three columns are not service stations that come after training. They are part of the decision before training.

Medical protects health context: injury, status, risk, return to work and information that determines load boundaries. Mental protects factors that influence readiness, focus, stress and the player's ability to respond to a demand. Nutrition connects intake, recovery and nutritional support with real load and schedule.

When wellness is reduced to a warning that stands aside, the coach sees it only after a problem has become large. When wellness is part of the shared workspace, it can affect the moment, dose, form and order of work before a decision is locked.

This does not mean wellness decides instead of the coach. It means the coach does not decide without the player's real state.

Administration: What Makes Work Possible

Management is the operational reality of the club. This is where schedule, teams, memberships, budget, equipment, contracts and licences, travel, documents, ordering and safeguarding live.

Administration is often called "back office" because it is not visible on court. But it determines whether the court exists, whether the hall is free, whether equip-

ment is ready, whether travel is organised, whether documents follow rules and whether a plan can be delivered without improvisation.

That is why administration is not a secondary module. It is one of the reasons SIMBION OS needs to be an operating system rather than only a sports application.

If training requires equipment that has not been ordered, if travel removes recovery time, if a contract or licence changes player availability or if a schedule opens a different window for work, this is no longer an administrative footnote. It is part of the sporting decision.

Kernel and Final: Shared Flow Without Dispersion

Simbion, RAG and Studio keep knowledge, assistance and creation connected. Final protects confirmed output. They do not replace professional categories; they prevent the workflow between them from being lost.

In a real day, one decision may begin with a player, move through training, check wellness, open an administrative condition and end as a document, working card, schedule or clip. Without a shared kernel, that path becomes a series of disconnected files. With a shared kernel, it remains one working story.

> **Wellness and administration are not around sports work. They are inside sports work.**

That is why SIMBION OS does not divide categories into “main” and “secondary”. Player, team, training, wellness, administration, knowledge and output exist so that the club can make a decision that has meaning and deliver it in the real day.

The next chapter connects all categories through one flow: from information to confirmed output.

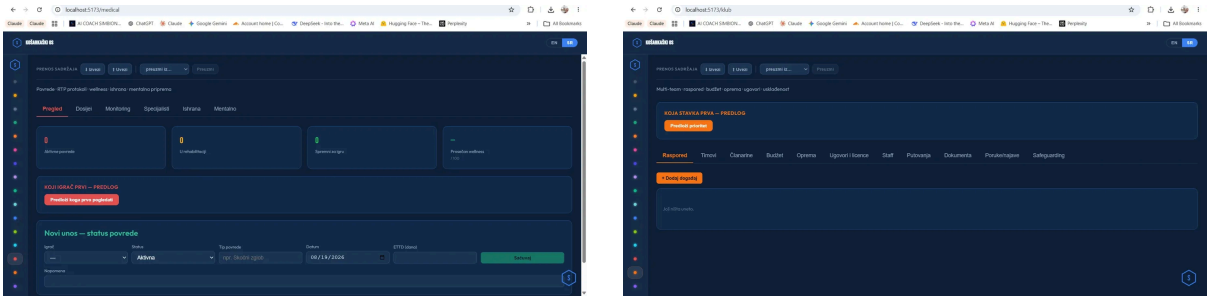


Figure 6. Medical and Management: wellness and administration are equal conditions of real work feasibility.

5 From Information to Confirmed Output

A system is not proven by the number of columns it has. It is proven by whether it can help when a club has a real task.

The most important SIMBION OS question is therefore simple: how does a decision move from information to work that is confirmed, understandable and deliverable?

The answer is not one rigid line that every user must follow. Sports work changes. Sometimes a decision begins with a player; sometimes with a team, a game, the schedule, wellness or an administrative constraint. But regardless of the starting point, the system must allow relevant information to become connected before output becomes final.

The main OS flow therefore looks like this:

> **Player or team → state, diagnostics and statistics → Simbion and RAG for context → planning and training for decision → Studio for working content → wellness and administration for feasibility → Final for confirmed output.**

This flow is not a prescription. It is a map of relationships.

First Step: The Question Gains Its Real Subject

Every quality process begins with a question: who does the decision affect?

Sometimes the answer is one player. The player may need a return-to-work plan, individual training, load adjustment or a visual explanation of a task. Sometimes the answer is the team. The club may be preparing a week with two games, moving into a new phase of the season or addressing a problem that affects the group.

When the system knows whether the work concerns a player or team, the question stops being abstract. It gains history, obligations, previous work, state and consequences.

This is the first way the OS protects work quality: it does not allow a plan to emerge without the subject to which it relates.

Second Step: State Does Not Remain in One Corner of the System

Diagnostics, statistics and wellness provide three different views of state.

Diagnostics describe a finding and profile. Statistics can show a pattern of performance, load or situation that requires attention. Wellness determines whether the plan is deliverable in the real day. None of those angles is sufficient alone. Together, they form the picture from which a priority can be set.

The system does not promise that every number will automatically produce an answer. It protects the availability of a number, observation, status and schedule in the same working moment.

This is where Simbion and RAG have their function. Simbion helps frame the next question. RAG returns relevant content and previous context. Instead of the staff searching through files for information, the system brings the information a decision needs closer.

Third Step: A Decision Becomes Planning and Training

When a priority becomes clearer, work moves into Planning and Training.

Planning places a decision in time: in the micro-, meso- or macro-cycle. Technique, Tactics, Physical Preparation and Drills provide usable content. Training turns that content into a concrete session.

It is important to understand that the system does not create a plan simply because it has much material. Content gains meaning only when it is connected to player or team, state, objective, schedule and the real conditions of delivery.

If the team is in a week with two games, the decision about dose cannot be the same as it would be in a broader training period. If wellness shows a constraint, the plan cannot act as if that constraint does not exist. If administration confirms a shorter slot or missing equipment, training must take a form that can be delivered.

The system does not separate planning from reality. It tries to keep the connection between what is desirable and what is possible.

Fourth Step: Studio Turns a Decision into Working Material

Not every output is text or a table. Sometimes a coach needs a court. Sometimes a player needs a short clip. Sometimes the team needs a diagram, working card or clear visual message.

Studio takes a decision that already has context and turns it into material that can be shown, explained or trained. This is the moment when knowledge leaves the form of a note and becomes something that can live on court, in the locker room, in a meeting or in individual work.

Good visual material is not decoration. It shortens the path between a professional idea and the user’s understanding. But, like every other output, it must remain connected to the reason it was created.

Fifth Step: Wellness and Administration Check Whether the Decision Can Live in Reality

Before work takes final form, the system returns the decision to the real day.

Medical, Mental and Nutrition examine readiness, recovery and work boundaries. Management checks schedule, space, equipment, travel, documents and other resources. This check does not exist to block the coach. It exists to make the decision genuinely deliverable.

If feasibility changes, the system does not need to destroy the entire work. It can return the decision to an earlier step, adjust the dose, change the form, protect the objective and produce new output. That is the difference between a rigid plan and an operating system.

A rigid plan is defended because it was written once. An operating system protects the reason for the plan while adapting it to the real day.

Sixth Step: Final Confirms the Output

Final is the moment of confirmation. It is not only a folder containing the latest version. It is where the staff know that output is ready for use.

Confirmed output may be training, schedule, plan, working card, visual task, clip, report or document. Its value lies in the fact that it did not emerge by accident and can be traced back to the context from which it came.

In this way, the system closes the loop without closing the work. When new information, a new game, a status change or another request appears the next day, output can be reopened, checked and adapted. The system does not force the club to begin again. It enables the club to continue from the real point at which it stands.

> **A good decision is not only one that looks correct on screen. A good decision can be delivered on court, in the schedule and in the real day of the club.**

This is the flow SIMBION OS protects: from information, through context and professional work, to output the club can use.

The next chapter shows that flow through two practical days: a competition day and a changed plan when real constraints modify the original intention.

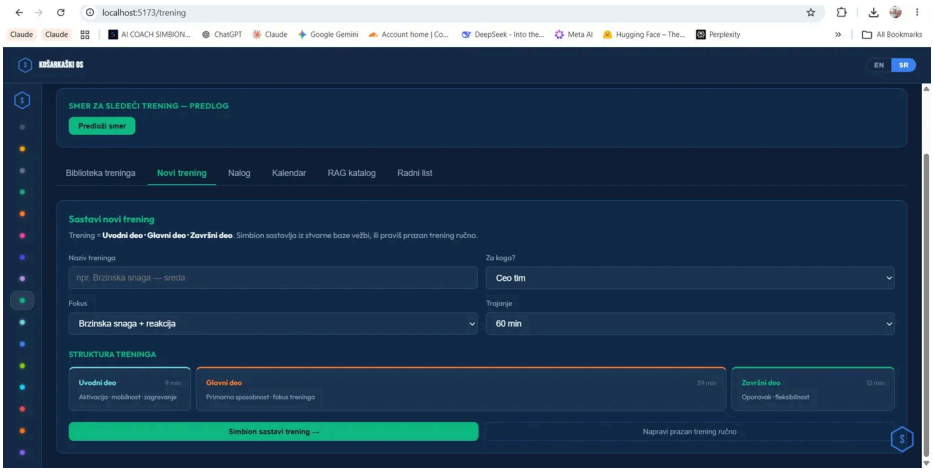


Figure 3. Training turns priority, structure and context into a concrete session.

6 A System That Serves the Club’s Real Day

An operating system is not judged by how good it looks when everything is calm. Its value becomes visible when the day is not ideal.

In sport, a plan is almost never delivered exactly as it was first written. A game brings contact, travel shortens recovery, the schedule changes a time slot, a player reacts differently than expected, the hall is unavailable, equipment is delayed, part of the team has another task or the staff receive new information only hours before work.

SIMBION OS is therefore not built to protect a plan from change. It is built to protect the reason for the plan while change happens.

First Scenario: Competition Day

The team enters game day. The plan has already been created, but that day is not simply the final page of the plan. It is a moment when several types of information need to be read together.

The coach opens Team to see the group and the day’s schedule. Wellness is checked from the working palette. Medical provides information on a player who had contact in the previous session. Nutrition confirms that the intake plan matches game time. Mental adds context on a player returning from a demanding week. Management checks travel, equipment and the time frame before arrival at the hall.

None of those pieces of information creates the decision alone. Together, they determine how the decision will be delivered.

The coach then opens Planning and Training. The work can be shortened, the order changed, activation adapted, an individual task checked or a clear message prepared for the group. If necessary, Studio prepares a court or short visual material for a meeting.

When the decision is confirmed, Final stores the material actually used that day: the work plan, message, visual, schedule or another concrete output.

This is not a story about a system automatically running a game. The coach and staff make the decision. The system protects the visibility of player, team, wellness and organisational context before that decision becomes work on court.

Second Scenario: The Plan Changes Because Reality Changed

Imagine that a training session with a larger dose has been prepared for the following day. Three new pieces of information arrive: the hall is available for less time, one player has a medical constraint and travel timing has moved.

In disconnected tools, staff need to open several messages, check multiple tables, find the latest version of training and manually explain the change to everyone involved. One detail is easy to lose. It is even easier to retain the original plan simply because change is complicated.

In SIMBION OS, the current training session remains the starting point. Management brings the shorter slot. Medical sets the boundary for the player. Planning shows where the session sits in the larger cycle. Wellness and schedule explain why the dose changes. The coach then adapts work in Training so that the objective is not lost but the form becomes deliverable. If a new court or clip is needed, Studio prepares it. Final stores the new confirmed version.

Most importantly, the change does not begin from a blank page. It begins from existing work and context that are already known.

> **A system is not good because it prevents change. It is good because change does not destroy the meaning of work.**

Third Scenario: An Individual Player Within a Team Frame

A player may have an individual plan, but never trains outside the team, schedule and the real day of the club.

For example, a player returns after a short absence. Medical confirms boundaries. Physical Preparation determines the route back. Tactics show the team obligations that are coming. Nutrition and Mental provide wider readiness context. Management determines available resources and time slots. Training turns all of that into concrete work.

If each piece of information were isolated, the player could easily receive several good but uncoordinated plans. When they are connected, the staff can create one decision that knows what the player can do, what the team needs and what the day really allows.

This is particularly important when speaking about individualisation. Individualisation is not giving everyone a different sheet of paper. Individualisation is giving every player work that makes sense in relation to profile, team task and real conditions.

Fourth Scenario: From Meeting to Working Material

Staff often have a good idea in a meeting. The problem appears when the idea does not become material that everyone understands and can use.

SIMBION OS allows a meeting to become more than conversation. Team or tactical context can become a plan. A plan can become a working card. A working card can become a court, diagram or clip in Studio. Studio can create confirmed material in Final. Final can provide content to a player, group, coach or another staff member.

In this way, the trace between idea and court is protected. When the club later checks what was done, why it was done and what should change, it does not need to reconstruct the decision from memory.

The System Leaves Space for People

One thing remains unchanged in every scenario: the human being is the decision authority.

The system can connect, flag, locate, prepare, organise and retain. It cannot take responsibility for professional judgement without a professional person who sees the player, team, situation and consequences of a decision.

The purpose of SIMBION OS is therefore simple. The club should not lose energy to the dispersion of context. Professional people should have more room for what no one else can do instead of them: assess, communicate, lead and decide.

> **The system serves the club's real day when it does not remove people's authority, but returns their overview of work.**

This completes the essential OS circle: from the need for one working environment, through the palette and kernel, to categories and a practical flow a club can use.

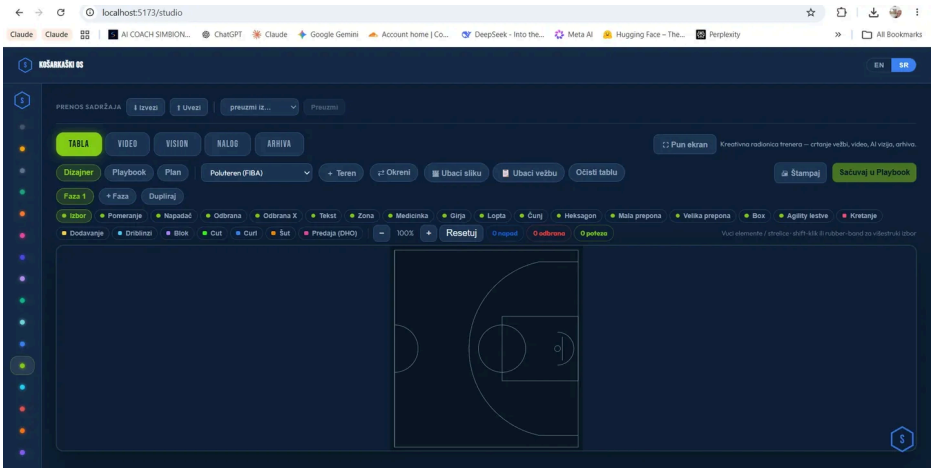


Figure 4. Studio turns a working idea into a court, diagram and visual material.

7 The System Does Not Replace Responsibility. It Makes It Visible.

SIMBION OS was not created because sport needs another platform. It was created because modern work can no longer be managed well through disconnected files, messages, tables and private notes.

When player, team, diagnostics, planning, training, wellness, administration, content and final output are separated, professionals spend too much energy transferring context. Information arrives late. Versions multiply. A decision loses the reason it was made. And when the real day changes, the club often chooses between two poor options: follow the old plan blindly or start from scratch.

SIMBION OS offers a third path.

It protects work inside one environment. It allows the user to open the tool that is needed without leaving previous context. It connects knowledge, content, planning, state and the real conditions of work. It helps information become confirmed output without losing the connection to player, team and the reason for the decision.

But the system does not replace responsibility. It makes responsibility visible.

The coach still leads the work. The staff still carry the decisions of their disciplines. The club still determines priorities, people, resources and working culture. SIMBION OS does not promise that one technology will solve the complexity of basketball. It ensures that professional knowledge does not remain locked in separate places.

Its purpose is therefore simple: quality work should not be lost between good intention and the real day.

> **You do not open a programme. You enter a workspace.**

If the player is a priority, the system needs to know the player's context. If the team is a priority, the system needs to connect people, plan and obligations. If training is a priority, the system needs to check whether it is deliverable. If wellness is a priority, it needs to be part of the decision rather than a footnote. If administration changes the reality of the day, it needs to be visible before the plan becomes final.

That is an operating system for sports work.

This book began with the player and physical preparation. It then showed how SIMBION turns methodology into personalised training. Part II widened

the perspective: the quality of work for one player, one team or an entire club depends on whether professionals can work with the same picture.

FPK I provided methodology. FPK II provided a court translation. SIMBION organised individual physical preparation. SIMBION OS connects the broader reality of sports work.

The question is now practical.

Do not try to run a club through disconnected files, messages and tables. Begin with the real day: player, team, wellness, training and organisation need to be in the same place before a decision is made.

The system does not then become a replacement for people.

It becomes the space in which good work can remain good when circumstances change.

> **The number of Columns and sub-columns is not the complexity of the work. It is the breadth of work now held in one place. Once you begin using SIMBION and SIMBION OS, each next step becomes a simple part of everyday work: open what you need, use it and continue from where you stopped.**

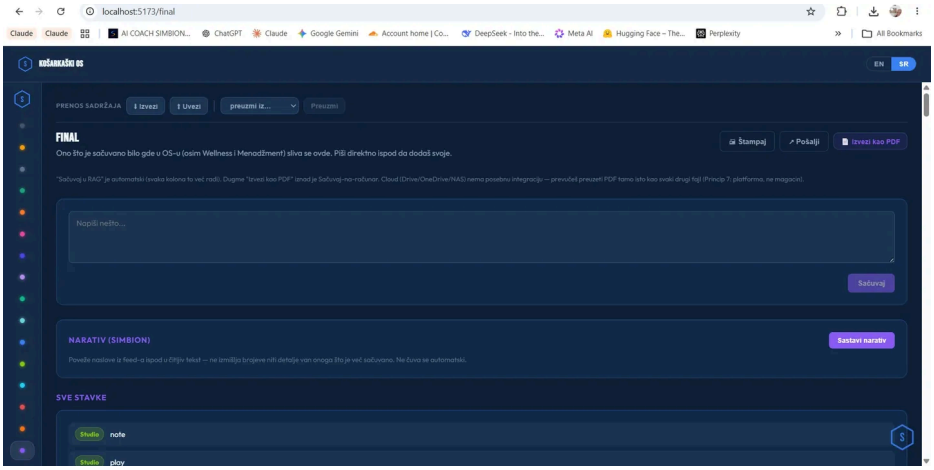


Figure 5. Final keeps confirmed output ready for further work, sending or printing.