

IMF as the Operating System of CreatiCampus Projects: A Process Framework from POAC to the Viewboard and the Creative-Campus Project Lifecycle

Eiser Aaron Beldiq^{1,*}, Greisy Jacqueline², Abigail Husain³

¹*Rey Incorporation, United States*

²*iLearning Incorporation, Estonia*

³*Mfinitee Group, South Africa*

*Corresponding author: eiseraaron_qrey77@rey.zone

Abstract

Creative-campus projects, in which students, lecturers and partners turn ideas into prototypes, creative startups, papers and events, often stall between enthusiasm and verified output. This paper proposes that Indicator Measurement Factor Analysis (IMF), a spreadsheet-based project method used at an Indonesian private university since 2019, can serve as their operating system. We first review how IMF has been described in a 2019 blog post, a 2020 conference paper, the literature citing that paper and the structure of live IMF spreadsheets. The published descriptions are brief and the citing works rarely engage with the method, but practice has added countdowns, closing rules, lead and task owners, evidence links, opening and closing narratives and recognition. Using a design-oriented approach, we derive six requirements for creative-campus projects from the creative-economy policy context and build a process framework, IMF-OS, that runs planning, organising, actuating and controlling as one repeating mission cycle. IMF-OS has five layers and five rules, a lifecycle of chained missions from idea to prototype to startup, paper or event with explicit gate decisions, a role model built on lead and task owners, a version cadence with adaptable rituals and safeguarded recognition, and an operational privacy procedure. A worked mission template, explicitly illustrative, shows the system in use. An analytical walkthrough and a traceability table show how the framework meets each requirement and which elements are documented, observed or newly proposed. No empirical results are claimed; the framework is offered for piloting.

Keywords: Indicator Measurement Factor Analysis; IMF; process framework; project operating system; POAC; Viewboard; creative campus; creative economy; project lifecycle; recognition rituals

1. Introduction

1.1 The Gap Between a Good Idea and a Finished Output

Every university that wants to contribute to the creative economy faces the same practical problem. Ideas are plentiful, but finished outputs are scarce. A student team sketches an app that animates traditional textile motifs, a lecturer imagines a community festival that could showcase local crafts, and a research group plans a paper on creative entrepreneurship. Weeks later the app is half built, the festival has no partners and the paper is still an outline. The ideas were good; what failed was the routine that should have carried them from intention to evidence.

This problem is not new, and it is not unique to creative work. Project-management research in Indonesia shows that even professional organisations struggle to deliver on time: one information-technology consultancy finished twelve of fourteen projects late over three years and needed a maturity assessment to understand why [1]. Agile methods have spread quickly, but their adoption

in Indonesian projects is hampered by cultural constraints, limited training, unclear guidelines and insufficient management support [2]. Universities add their own difficulties. Grant management still relies on fragmented monitoring, limited real-time information and systems that are not designed around their users [3], and manual or outdated platforms leave lecturers with unclear progress tracking, slow updates and avoidable stress [4]. Performance parameters imposed on faculty members can themselves become a source of pressure that harms teaching and learning [5].

Creative-campus projects sharpen these difficulties. They are typically short, they involve people who are not each other's managers, and they must deliver outputs that someone outside the team can recognise: a working prototype, a registered copyright, a submitted paper, a pitch deck, an event that actually takes place. They also depend heavily on voluntary energy. Research on creative entrepreneurs emphasises iterative, user-centred work that conventional business models support poorly [6], and education studies show that creative and entrepreneurial programmes have a significant influence on whether students become creative entrepreneurs at all [7]. A campus that wants to grow creative talent therefore needs a way of running projects that is light enough for students, rigorous enough for partners and motivating enough to keep teams going.

1.2 Why Turn to IMF

Indicator Measurement Factor Analysis (IMF) is a candidate for that role. It was introduced at Universitas Raharja in 2019, in a blog post that presented it as a method for completing projects through the four classical management functions of planning, organising, actuating and controlling (POAC), and in a 2020 conference paper by Sudaryono, U. Rahardja and N. Lutfiani that described it as “a simple, low cost implementation tools that facilitate effective POAC control monitoring and implementation in project management” (J. Phys.: Conf. Ser. 1477, 032023, doi:10.1088/1742-6596/1477/3/032023). In the paper, an IMF has two parts: a Viewboard, a header that shows a project's title, objectives, outcomes, contributors, version and due date; and a Content grid in which each contributor completes coded tasks with a percentage of achievement. Since then the method has been applied in spreadsheets for grants, publications, accreditation, teaching and events, as Section 2.4 describes.

What makes IMF interesting for creative-campus projects is not any single feature. Dashboards, task lists and percentage trackers are everywhere. It is the combination: a declared mission with countable outputs, a task grid with named owners and evidence, a live measure of achievement and time, and a set of rituals that open and close each mission. This combination resembles what an operating system does for a computer. An operating system does not write the user's documents; it schedules work, allocates resources, keeps records, shows status and enforces a few rules that let many programs run without chaos. We argue that IMF can play the same role for a portfolio of creative-campus projects.

1.3 Aim and Contribution

The aim of this paper is to propose and specify IMF as the operating system for creative-campus projects. Following the brief that motivated this series, we proceed in three steps. First, we review how IMF has been described and used by others. Second, we propose a new operational idea and usage of IMF, which we call IMF-OS. Third, we argue that IMF-OS is an essential tool for building creative-campus projects and show how it works in practice through roles, cadence, rituals and a worked mission template.

The paper makes three contributions. It offers a compact review of IMF's published descriptions and practice that is oriented to use rather than to definition; a companion concept analysis in this working-paper series addresses the definitional questions [8]. It provides a process framework that maps POAC onto IMF's components and onto the lifecycle of a creative-campus project, from idea to prototype to creative startup, paper or event. And it supplies an illustrative mission template that a campus team could copy and adapt. The framework is a design proposal. We

report no empirical results; testing is left to pilots and to the measurement instrument that a later paper in this series develops [9].

1.4 Structure of the Paper

Section 2 reviews how IMF has been described and used. Section 3 explains the design approach and the requirements that the framework must meet. Section 4 presents the IMF-OS architecture and its mapping to POAC. Section 5 describes the creative-campus lifecycle as a chain of missions. Section 6 defines the roles, cadence, rituals and recognition. Section 7 presents the illustrative mission template. Section 8 evaluates the framework against the requirements and against related methods, and Section 9 discusses implications and limitations. Section 10 concludes.

2. How IMF Has Been Described and Used

2.1 The Founding Descriptions

The earliest public description of IMF is a blog post titled “Why IMF?”, dated 26 June 2019, written by N. Lutfiani on the blog of U. Rahardja (archived at <https://web.archive.org/web/2024/https://untungrahardja.ilearning.me/2019/06/26/why-imf/>). Written in Indonesian, it credits the innovation to U. Rahardja and describes IMF as a method for completing information-technology projects by focusing on the implementation of POAC, which it calls the basis of management science. The post argues that many people claim to use POAC but few succeed, because applying it requires tools that are often costly. Its practical instruction is short: every project must have objectives (Tujuan), outputs (Luaran) and contributors, whose performance is measured with a spreadsheet the post calls RinfoSheet. It names three tools, RinfoSheet, vasTmind and RinfoCal, and three example IMFs for a research plan, a business incubator and a national research grant. It notes that IMF is also the abbreviation of “Impossible Mission Force”, an allusion to the film series about near-impossible missions, and hopes the name will give people a spark of motivation.

The 2020 conference paper presents IMF to a scientific audience. It frames the problem in similar terms: POAC is widely accepted, but many organisations do not know how to implement it, are confused by the variety of applications and cannot afford the tools. It proposes a combination of spreadsheets and the vasTmind mind-mapping method as a simple solution and describes the two parts of an IMF. The Viewboard is “the main key to achieving IMF”, a header that displays all the information a leader needs “without having to scroll down”, using functions such as HLOOKUP, VLOOKUP, AVERAGE, SUM, MAX and MIN. The Content “contains the entire project work process that consists of indicators to be measured”, and in it “each contributor can complete the task with a percentage achievement and a unique code on each project”. The paper reports that “version changes run every month”, shows a screenshot of an IMF for a research master plan, and claims that IMF produces effective results in “communication, collaboration, critical thinking, competition, and creativity of teamwork”. It does not report data on these effects.

Three features of these founding texts matter for the present paper. They describe IMF as a way of implementing POAC rather than as a new theory of management. They treat the mission, with its objectives, outputs and contributors, as the unit of work. And they present IMF as a low-cost, spreadsheet-based practice rather than as a software product. None of them, however, says how IMF should be used across the stages of a project, who should do what, or how often the cycle should turn. Those are the questions an operating system must answer.

2.2 Antecedents and Neighbours in the Literature

IMF did not appear in isolation. Its authors’ community had already written about Viewboards as compact information displays; one study, for example, developed a Viewboard to strengthen the credibility of an association’s website as an information medium [10]. Later work from the same

community designed grant-monitoring platforms that emphasise real-time tracking, structured reporting and transparency of process [3,4], which are among the functions that IMF performs in spreadsheet form. A short community-service study by U. Rahardja promoted the fundamentals of agile development as a way of building the technology skills of young people [11], indicating that iterative methods were part of the same intellectual environment.

Beyond this community, Indonesian studies of project and task management show the range of tools that IMF must sit beside. Task-management systems have been designed to integrate task recording, assignment and reporting so that the performance of divisions can be monitored [12]. Dashboards have been designed to monitor teacher performance assessments [13], and mobile applications have been built to monitor student final projects with reminders for students, supervisors and examiners [14]. Earned-value management has been used to evaluate cost and schedule performance in village cooperative construction projects [15], and maturity frameworks have been used to diagnose why projects are late [1]. Studies of management and monitoring stress that strong monitoring allows organisations to assess the effectiveness of their initiatives and identify areas for improvement [16]. Each of these tools addresses part of what IMF does; none combines a declared mission, coded tasks, live achievement and ritual in the way IMF does.

2.3 How the Citing Literature Uses IMF

A natural place to look for others' interpretations of IMF is the literature that cites the 2020 paper. OpenAlex lists 49 citing records, which reduce to 46 distinct works once duplicates are merged. We read the sixteen open-access works among them that appear in the publisher network most closely connected with the method's home institution, and classified the sentence in which each cites IMF. None of the sixteen describes, applies, critiques or extends IMF. Two cite it next to statements about management, governance or time management; three cite it next to statements about data processing or research method; and eleven cite it in sentences with no evident link to project management or measurement. In half of the reference entries the paper's first author is omitted. These findings, which are reported in more detail in the companion concept analysis [8], mean that the citing literature does not yet offer an independent interpretation of IMF. Its value as evidence of use is low.

2.4 IMF in Practice

The richest evidence of how IMF is understood comes from the method's practice. We examined the structure, but not the personal content, of live spreadsheets titled "IMF ..." that are owned by or shared with the method's originator, in read-only mode. The sheets cover research grants, publication campaigns, paper-processing batches, teaching programmes, internal events and multi-team sprints. Their structure is remarkably consistent, and it goes well beyond the founding texts. Because the sheets use Indonesian and IMF-specific terms, Table 11 in Appendix A gives a glossary with each term's literal meaning and its function in the framework.

The Viewboard has become a dense header. It typically shows the IMF's title, a purpose statement (Tujuan), output counts (Luaran Target, computed by counting the coded tasks in the planned section; Luaran Extra, counting tasks in an extra section; and Luaran Total), start and due dates, a countdown formula that displays the number of days left or "Finish", a done date with the number of days finished early, a closing flag that reads "CLOSING" or "NOT CLOSING", and an achievement percentage (Capaian) computed as the sum of task achievement divided by the target or total. Files with several teams add roll-ups of target, completed and remaining items per team, and cells that name the most valuable lead and task owners ("MVP LPIC" and "MVP PIC"). Links to a guide, a shared folder and a submission form are common.

The Content has become a structured task ledger. Each row carries a unique code, often called a Lovecode and formed from the team or project name and a number; a task description; a checkbox; an achievement cell; the person in charge (PIC) and, in multi-team files, a lead PIC (LPIC); and a

notes cell that often contains a link labelled “Disini” (“here”) to the evidence. Rows are grouped into planned work (“To Do”) and work beyond the plan (“Extra Job”). Domain-specific IMFs add columns for stages: a publication IMF, for example, tracks draft, citation targets, journal category, submission, letter of acceptance, payment, web publication and indexing.

Two further features are cultural. Many Viewboards contain formula-generated sentences that read the live numbers back as prose. An opening sentence beginning with “Bismillah” declares the team’s intention to deliver a number of outputs between two dates; a closing sentence beginning with “Alhamdulillah” reports the percentage achieved, often calling it a token of love (tanda cinta) from the team, and states how many days early the work was finished. And several IMFs recognise contributors: through most-valuable-contributor cells, through rankings by achievement and, in one gamified IMF seen only as a search snippet, through leagues and medal counts per team. No sheet we examined states a formal reward rule, and neither founding text mentions rewards.

Versioning varies. The 2020 paper speaks of monthly versions; one multi-team IMF uses a new tab for each sprint of about two weeks; others carry a static version number. Table 1 summarises the sources and what each contributes to the framework.

Table 1. Sources reviewed and their contribution to the framework

Source	What it is	What it contributes	Limits
Blog post, June 2019	Indonesian-language post “Why IMF?” on the originator’s blog (archived copy)	Purpose (POAC implementation), the mission as unit of work, objectives-outputs-contributors, tools, the “Impossible Mission Force” motivation	Short; no roles, cadence or lifecycle
Conference paper, 2020	Open-access paper introducing IMF to a scientific audience	Viewboard and Content, formulas, unique task codes, monthly versions, claimed teamwork effects	No data on effects; one example shown
Citing literature	46 distinct works citing the 2020 paper; 16 open-access works read and coded	Evidence of how others interpret IMF	No work describes, applies or extends IMF
Practice sheets	Structure of live spreadsheets titled “IMF ...” read-only; no personal data	Countdown, closing flag, Luaran Target/Extra/Total, LPIC and PIC, evidence links, opening and closing sentences, recognition cells, versions	Few sheets examined in depth; others seen in part
Related literature	43 recent studies on project and task management, incubation, creative entrepreneurship, motivation and campus collaboration	Requirements, lifecycle content, comparison methods [12, 17, 18]	None tests IMF

2.5 What the Review Implies

The review yields four implications for an operating-system reading of IMF. First, the unit of work is the mission, and practice has made the mission bounded: it has a start, a due date, a countdown and a closing flag. Second, roles are already differentiated in practice, with a lead owner and task owners, even though the founding texts speak only of contributors. Third, the cycle is ritualised: missions open and close with narrative sentences and, in some cases, with recognition. Fourth, what is missing is guidance on how missions follow one another, how many people should hold which role, how often the cycle should turn and how a campus can run many IMFs at once. Table 2 sets out, for each element of IMF, what the founding texts say, what practice shows and what an operating system would need to add.

Table 2. Elements of IMF in the founding texts and in practice, and what an operating system must add

Element	Founding texts	Practice	What IMF-OS adds
Mission	Every project has objectives, outputs and contributors	Bounded by start, due date, countdown and closing flag	Mission chaining across lifecycle stages; portfolio view
Viewboard	Header with title, objectives, outcomes, contributors, version, due date	Purpose, Luaran Target/Extra/Total, countdown, done date, closing flag, achievement, roll-ups	A fixed set of essential fields and the two-number rule
Content	Tasks with percentage achievement and unique code	Coded rows, checkbox, owner, evidence link, planned and extra sections	Evidence rule; extra tasks recorded separately
Roles	Contributors	Task owner (PIC) and lead owner (LPIC); most-valuable cells	Full role model with mentor, partners and steward
Cadence	Versions change every month	Sprint tabs of about two weeks, or static versions	Default two-week version with four moments; monthly portfolio review
Rituals	Not mentioned	Opening and closing sentences computed from the Viewboard	Function separated from expression; adaptable wording
Recognition	Seeing best and lowest contributor (blog)	MVP cells, rankings, early-finish lines, medals in one case	Four safeguards; criteria declared at opening

3. Design Approach

3.1 A Design-Oriented Framework Study

This is a design-oriented conceptual study. Its product is a process framework, not a test of hypotheses. We followed five steps. First, we reviewed the descriptions and practice of IMF, as reported in Section 2. Second, we derived requirements that any operating system for creative-campus projects should meet from the creative-economy policy context and from the literature on campus projects, incubation and motivation. Third, we constructed the framework by mapping POAC onto IMF's components and arranging those components into layers, a lifecycle, a role model and a cadence. Fourth, we instantiated the framework in an illustrative mission template. Fifth, we evaluated the framework analytically, by walking through each requirement and by comparing the framework with related methods.

The approach borrows from design-thinking practice, which moves from empathy and definition to ideation, prototyping and testing [6], and from development research that designs a product, checks its feasibility and refines it [19,20]. It differs from empirical design research in one important respect: the testing step is analytical. We do not claim that IMF-OS improves outcomes; we claim that it is coherent, that it follows from the evidence of IMF's use and that it meets the requirements well enough to be worth piloting. Because the founding texts are brief, we distinguish throughout between elements that are documented in those texts, elements observed in practice and elements that we propose. Section 8.3 and Table 10 trace every element of the framework to its source and label its status, so that readers can see exactly what is new.

3.2 The Creative-Campus Context

By a creative-campus project we mean a bounded effort, led from a university, in which students, lecturers and partners turn a creative idea into an output that can be recognised outside the team. The CreatiCampus working-paper series is concerned with how universities create, nurture and scale creativity, creative talent and intellectual property for Indonesia's creative economy. Three

features of the current policy context shape the requirements for such projects.

The first is the legal framework. Law No. 24 of 2019 on the Creative Economy commits the government to facilitating financing schemes based on intellectual property and to facilitating the recording of copyright and the registration of industrial property for creative actors, and Government Regulation No. 24 of 2022 allows intellectual property to serve as collateral when the applicant can show a certificate or recording letter and related agreements. Campus projects that hope to benefit must therefore produce documented, attributable outputs.

The second is the master plan. Presidential Regulation No. 37 of 2026 on the Creative Economy Master Plan 2026–2045, signed in July 2026, sets eight priority programmes that include creative-economy data, talent, intellectual property and synergy among six actors, the so-called hexa helix of government, academia, business, community, media and financial institutions. Penta-helix and hexa-helix collaboration is already a theme of campus community service [21, 22], and studies of creative hubs show that micro entrepreneurs co-create value through networks rather than within single organisations [23].

The third is the calendar. Creative-campus projects often run to deadlines set by others: grant calls, competitions, exhibitions and policy events. The fifth World Conference on Creative Economy (WCCE) is scheduled for 21–23 October 2026 in Jakarta under the theme “Inclusively Creative: Collective Continuity”; an earlier paper in this series reads that theme as an Inclusive Creative Campus framework of access, capability, care and continuity [24]. Events of this kind create windows in which a prototype, a showcase or a paper can reach a wide audience, but only if it is ready on the day.

3.3 Design Requirements

From this context and from the literature reviewed above, we derive six requirements for an operating system for creative-campus projects.

- R1.** Countable, evidenced outputs. The system must turn creative intentions into outputs that can be counted and verified, so that teams, mentors and partners agree on what has been done and so that outputs can later support intellectual-property applications, incubation and publication [17, 25].
- R2.** Multi-actor accountability without hierarchy. The system must make each contributor’s part visible across organisational boundaries, because creative-campus teams combine students, lecturers and external partners who do not report to one another [26, 27].
- R3.** Alignment with external deadlines. The system must translate deadlines set by funders, competitions and events into a shared sense of time within the team, with a clear moment at which the work is declared finished.
- R4.** Low cost and a low skill threshold. The system must run on tools that students and lecturers already have and must be learnable in an afternoon, because the users are not professional project managers and campus budgets are limited.
- R5.** Inclusive, sustainable motivation. The system must sustain voluntary effort through meaning and recognition while avoiding the stress that performance metrics can create [5, 28].
- R6.** Aggregatable, protected records. The system must produce records that a campus can combine across projects to report creative outputs, consistent with the master plan’s emphasis on creative-economy data, while protecting the personal data of contributors under Law No. 27 of 2022 on Personal Data Protection.

These requirements serve two purposes. They guide the construction of the framework in Sections 4 to 7, and they provide the criteria for its analytical evaluation in Section 8.

4. The IMF Operating System

4.1 A New Operational Idea of IMF

The review suggests a new way of understanding IMF. The founding texts present it as a tool for implementing POAC, and the practice shows it as a family of spreadsheets. We propose to treat it as an operating system for collective work: a small set of records, routines and rules that every project runs on, regardless of what the project produces. We call this reading IMF-OS and define it as follows.

Operational definition. IMF is the operating system of a creative-campus project: a light, spreadsheet-based routine that runs planning, organising, actuating and controlling as one repeating mission cycle, in which a lead owner opens each version by declaring its purpose, countable outputs and time window on the Viewboard, task owners deliver and evidence uniquely coded tasks in the Content grid, the Viewboard computes achievement and time remaining continuously, and the team closes the version with a public account of what was achieved and who contributed before the next stage of the project begins.

The definition does not replace the founding descriptions; it operationalises them. “Simple” and “low cost” become a routine that runs on a shared spreadsheet. “POAC control monitoring” becomes a cycle that opens, runs and closes. “Contributors” become lead and task owners with distinct duties. And the practice features that the founding texts do not mention, such as the countdown, the closing flag, the evidence link and the opening and closing sentences, become part of the system rather than local decoration. The idea is also new in two respects. It treats successive missions as a chain that carries a project from idea to output, and it treats a campus’s many IMFs as a portfolio that can be read at a glance.

4.2 Five Layers

IMF-OS has five layers, shown in Figure 1. Each layer corresponds to a function that an operating system performs and to elements already present in IMF practice.

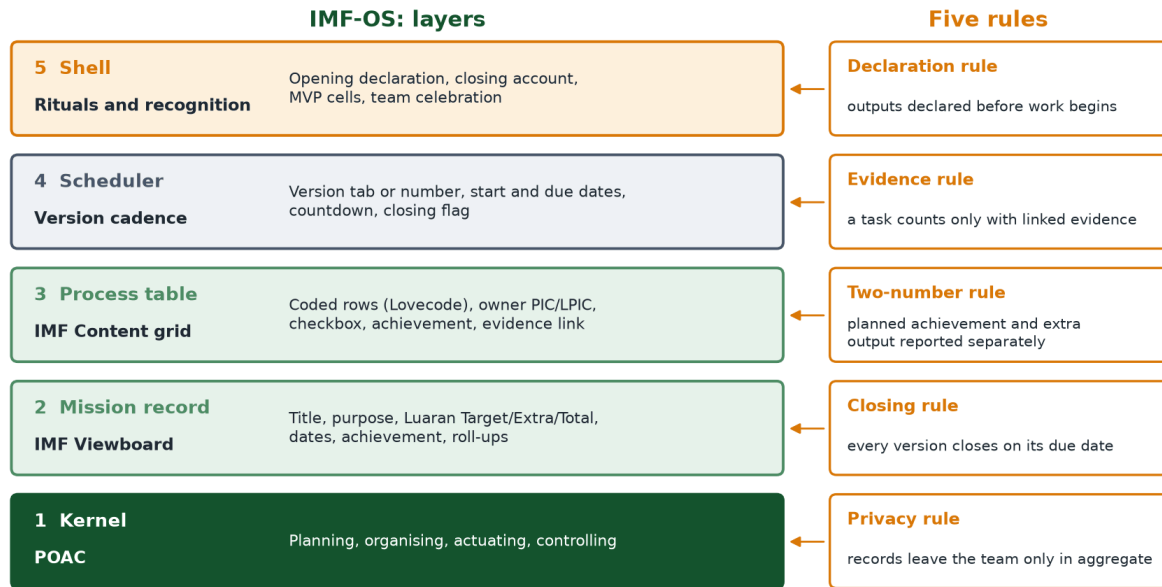
The kernel is POAC. It is the smallest set of management functions that every project must perform, and the founding texts are explicit that IMF exists to implement it. The kernel does not appear on the screen; it is the logic that the other layers express.

The mission record is the Viewboard. It is the system’s file header: the place where a mission’s identity, purpose, outputs, owners, dates and status are declared and computed. Its essential fields are the title, the purpose statement, Luaran Target, Luaran Extra and Luaran Total, the start and due dates, the countdown, the done date, the closing flag and the achievement percentage. A mission without these fields is not yet an IMF mission.

The process table is the Content grid. Each row is a process in the operating-system sense: a unit of work with an identifier, an owner, a state and an output. The unique code links the row to the mission and to its evidence; the checkbox and achievement cell record its state; the PIC and LPIC cells record its owner and lead; and the evidence link records what was produced. Grouping rows into planned work and extra work keeps the denominator of achievement honest.

The scheduler is the version cadence. It decides when a mission opens, how long it runs and when it must close. In practice this is the version tab or version number, the start and due dates and the countdown formula. The scheduler is what turns an open-ended list into a time-boxed mission.

The shell is the set of rituals and recognition through which people interact with the system. The opening and closing sentences, the most-valuable-contributor cells and the celebration of early finishes are the system’s user interface for meaning. They tell the team, in words, what the numbers mean.



Apps running on IMF-OS: idea, prototype, startup, paper and event missions

Figure 1. The five layers of IMF-OS: POAC as kernel, the Viewboard as mission record, the Content grid as process table, the version cadence as scheduler and rituals with recognition as shell, with the five rules that govern them.

4.3 Mapping POAC to IMF and to Creative-Campus Work

Neither founding text maps the four POAC functions one by one onto IMF's components. The practice, however, implies a mapping, which we make explicit in Table 3. Planning corresponds to the Viewboard declaration of purpose, outputs and dates; organising to the assignment of lead and task owners and the coding of tasks; actuating to the completion and evidencing of tasks in the Content grid; and controlling to the computed achievement, the countdown, the closing flag and the recognition cells. The mapping is our inference from practice and should be read as a proposal that the method's originator and users can confirm or amend. It is also a mapping of primary carriers, not a sequence. All four functions recur within every version: planning happens at each opening and again whenever the team re-plans at the mid-version check; organising continues when extra tasks are added and assigned; actuating runs throughout; and controlling happens continuously through the computed fields and deliberately at the mid-version check and the closing. What Table 3 shows is the element in which each function is made visible and recorded, which is why the cycle in Figure 3 is drawn as a loop rather than a line.

Table 3. Mapping of POAC functions to IMF-OS elements and to creative-campus work (our inference from practice)

POAC function	IMF-OS element	Typical failure it prevents	Creative-campus example
Planning	Viewboard declaration: purpose, Luaran Target, start and due dates	Vague outputs; no deadline	"Eight outputs, including a tested prototype and a showcase slot, by 20 October"
Organising	Coded Content rows with PIC and LPIC; partners named	Tasks nobody owns	Each build, test and document has one owner
Actuating	Task completion with checkbox and evidence link	Work done but undocumented	Design files and test records linked to rows

POAC function	IMF-OS element	Typical failure it prevents	Creative-campus example
Controlling	Achievement, countdown, closing flag, closing statement, recognition	Nobody notices time running out; no closure	Day-ten view shows 62.5 percent with four days left

The mapping matters for creative-campus work because each POAC function corresponds to a typical failure. Projects fail in planning when outputs are vague; in organising when nobody owns a task; in actuating when work is done but not documented; and in controlling when nobody notices that time is running out. IMF-OS addresses each failure with a specific field or routine, which is what distinguishes it from a general exhortation to plan better.

4.4 Records, Formulas and Rules

An operating system is defined as much by its rules as by its components. IMF-OS has five rules, each derived from practice.

The declaration rule says that a mission’s outputs are declared before work begins. Luaran Target is the count of planned tasks at the moment of opening; tasks added later are recorded as extra work. This keeps the achievement percentage meaningful and makes scope growth visible.

The evidence rule says that a task counts as achieved only when its evidence is linked. The “Disini” link in the notes cell is the practice form of this rule. For creative-campus projects, evidence is often the very material that will later support an intellectual-property application or an incubator review: design files, test records, recordings and drafts. The rule has limits that should be stated plainly. A link in a spreadsheet records that a file existed and who said they produced it; it is not legal protection, it does not prove ownership and it does not replace copyright recording or other registration with the competent authority. Its value is preparatory: it makes the documentation that such procedures require easier to assemble, and it reduces later disputes about who did what.

The two-number rule says that planned achievement and extra output are reported separately. In practice, some IMFs divide achievement by the planned target and others by the total, so that values above 100 percent occur. IMF-OS reports planned achievement as verified planned tasks divided by Luaran Target and reports extra output as a separate count.

The closing rule says that every version ends on its due date with a closing flag and a closing statement, whatever its achievement. A version that closes at 70 percent is a finished version with an honest record; a version that never closes is an open loop.

The privacy rule says that records leave the team only in aggregate. Within the team, names and individual achievement are visible because accountability requires them. Outside the team, the system reports outputs by type, mission and period, with contributor columns removed or pseudonymised, consistent with Law No. 27 of 2022. In operational terms, we propose five provisions. Access is tiered: team members, the mentor and the steward see the full mission record; partners see the Viewboard summary without individual achievement unless the people concerned agree; and anyone beyond the mission sees only aggregates. Contributors are informed when they join a mission, through a short privacy notice linked from the Viewboard’s guide, of what is recorded, who can see it and for what purpose. Use of individual-level records beyond the team, for example in research or formal evaluation, requires the explicit consent of the people concerned. Records are retained in identifiable form only for a period set by campus policy, after which contributor columns are pseudonymised or deleted. And contributors can ask to correct records about themselves. These provisions are design proposals; each campus must align them with its own data-protection policy and legal advice.

4.5 What the Metaphor Captures and Where It Breaks Down

The operating-system metaphor captures four things: that IMF is infrastructure rather than content, since it runs any project without dictating what the project produces; that it has a small set of records, routines and rules that every project shares; that it schedules work in bounded cycles; and that it presents status through a common interface. The metaphor breaks down in three places, and readers should not infer what it does not intend. First, nothing in IMF-OS runs automatically: every declaration, update, verification and closing is done by a person who chooses to do it, and the system works only as long as people find it worth their effort. Second, the “processes” in the Content grid are tasks, not people; contributors are agents with their own motives, and the framework is designed to support their judgement, not to control them. Third, an operating system enforces its rules mechanically, whereas IMF-OS relies on shared norms, mentors and stewards. The metaphor is therefore an aid to design and explanation, not a claim that projects can be managed like machines.

5. The Creative-Campus Lifecycle as a Chain of Missions

5.1 Mission Chaining

A creative-campus project rarely fits in one mission. An idea needs to be explored before it can be prototyped, and a prototype needs to be tested before it can become a startup, a paper or an event. IMF-OS handles this by chaining missions: each stage of the lifecycle is run as one or more versions of an IMF, and the closing of one version is the gate to the next. Figure 2 shows the chain. Chains also carry risk: a case study of an interlinked venture ecosystem in this series distinguishes links between working capabilities from chains of unfinished projects [29], and a valuation study in the same series treats such chains as staged, abandonable investments [30]; the stage gates of IMF-OS are the points at which a chain can be stopped.

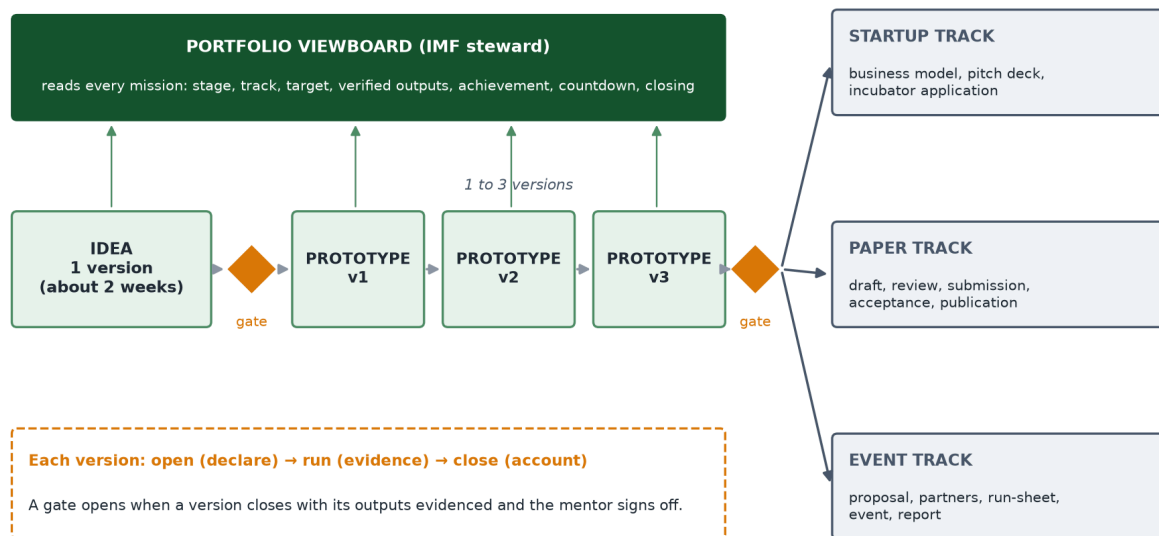


Figure 2. The creative-campus lifecycle as a chain of IMF missions: idea and prototype versions lead through stage gates to three output tracks (creative startup, paper and event), all read by a portfolio Viewboard.

The chain has three common stages and three output tracks. The idea stage turns a theme into a defined problem and a concept; the prototype stage turns the concept into a working artefact that has been tested with users; and the output stage turns the prototype into one of three recognisable outputs: a creative startup, a paper or an event. A project may follow more than one

track, and many projects will produce a paper as an extra output alongside a startup or an event.

5.2 The Idea Stage

The idea stage is a short mission, typically one version of about two weeks. Its purpose is to move from a theme to a defined problem and a concept that the team believes it can build. Typical Luaran are a problem statement, a summary of conversations with intended users, a concept sketch or storyboard and a team charter that names the lead owner and task owners. We recommend adding a fifth output: a short authorship and ownership agreement, signed by students, lecturers and any partner who will contribute creative material, that states how authorship of papers will be decided and who will hold rights in what the team creates. Mixed teams that settle these questions at the start avoid the disputes that most often arise when a prototype succeeds. Design-thinking studies describe the same movement through empathy, definition and ideation [6, 31], and lean-startup curricula frame it as the formulation of hypotheses to be tested [32]. The gate to the next stage is the closing of the idea version with all five outputs evidenced and a mentor's confirmation that the concept is feasible within the next window.

5.3 The Prototype Stage

The prototype stage usually runs for one to three versions. Its purpose is to build an artefact that works well enough to be shown and tested. Typical Luaran include a first build, a user test with a short record of results, a revised build and a provenance file that collects the design documents, source files and dated drafts that show who created what. Community-service programmes that train students in interface and experience design for digital products [33] and that move partners from prototyping to small-scale production [34] show the kind of skills involved, and studies of interdisciplinary learning-factory projects show how real-world project environments are being integrated into higher education [35, 36]. Incubator design work, such as three-dimensional interior design produced for a campus business incubator [37], illustrates how prototype work can itself become a documented output. The gate is a tested prototype with its provenance file linked. The provenance file does not itself protect the work, but together with the ownership agreement it gathers the documentation that a later copyright recording or licensing discussion would draw on.

5.4 The Output Stage and Its Three Tracks

The startup track turns a prototype into a venture that can enter incubation. Its Luaran typically include a business model, a pitch deck, an incubator application and, where appropriate, business registration. Studies of campus incubators have evaluated the practicality and effectiveness of new incubator models with experts and tenants [17], identified determinants of incubation success among incubator managers [38] and found that access to funding, experienced mentors and business networks are among the strategies that help technology startups grow [39]. Research on pitch decks shows how visual language and narrative convey value propositions to investors [25]. Each of these outputs can be declared as Luaran and verified by a link.

The paper track turns the project's knowledge into a publication. Its Luaran mirror the stages that publication IMFs already track: draft, internal review, submission, letter of acceptance and publication. Studies of scientific-publication management show that digital platforms and flexible editorial processes matter for efficiency [40], and studies of journal branding show the role of sharing and collaboration in dissemination [41]. A paper is often the most durable output of a creative-campus project, because it records what was learned even when the venture does not survive.

The event track turns the project into a public occasion: an exhibition, a showcase, a festival or a demonstration day. Its Luaran include a proposal, confirmed partners, a run-sheet, the event itself and a report. Community-service programmes show that socio-cultural public events are often run as ceremonies and underused as platforms for entrepreneurship, and that penta-helix

collaboration can turn them into such platforms [22]. Events are also where external deadlines bite hardest: an event that is not ready on the day cannot be postponed by the team alone.

Table 4 summarises the stages, their typical outputs, the evidence that counts and the gate that closes each stage.

Table 4. The creative-campus lifecycle as chained missions: stages, typical outputs, evidence and gates

Stage or track	Typical Luaran	Evidence linked	Gate to next stage
Idea (1 version)	Problem statement; user conversation summary; concept sketch; team charter; authorship and ownership agreement	Notes, interview summaries, sketch files, charter	All outputs evidenced; mentor confirms feasibility [6, 31]
Prototype (1–3 versions)	First build; user test; revised build; provenance file	Build links, test record, dated design files	Tested prototype with provenance file [33, 34]
Startup track	Business model; pitch deck; incubator application; registration where needed	Documents, submission receipts	Incubator decision or market test [17, 25]
Paper track	Draft; internal review; submission; acceptance; publication	Draft versions, submission and acceptance records	Submission or acceptance [40]
Event track	Proposal; confirmed partners; run-sheet; event; report	Letters, run-sheet, photographs, report	Event held and report filed [22]

5.5 Gate Decisions

A gate is a decision, not just a closing. When a stage’s final version closes, the lead owner and the mentor decide jointly, after hearing the team, among five options. The team proceeds when all gate outputs are evidenced and planned achievement is high; we suggest 80 percent as a default threshold. It proceeds with carry-over when planned achievement is moderate, which we suggest as 60 to 79 percent, and the missing outputs are not critical to the next stage; the missing outputs then become planned tasks of the next version and are counted in its target. It repeats the stage, with a new version and a revised declaration, when a critical output is missing, whatever the percentage. It pivots, returning to the idea stage with a new declaration, when the evidence, typically a user test, shows that the concept does not work. And it stops when the team and mentor agree that the project should end; the closing statement then records what was learned, and the paper track may still be used to share it. The thresholds are design choices, not findings: they are meant to make decisions explicit and comparable, and pilots should calibrate them. Every gate decision is recorded on the Viewboard with its date and reason, so that the mission chain shows not only what was produced but how the project was steered.

5.6 The Portfolio View

A campus that runs many creative-campus projects needs to see them together. IMF practice already contains the seed of this view: multi-team IMFs roll up target, completed and remaining items per team and name the most valuable lead and task owners. IMF-OS generalises this into a portfolio Viewboard maintained by a programme office or IMF steward. Each row of the portfolio is one mission; its columns repeat the mission’s stage, track, Luaran Target, verified outputs, achievement, countdown and closing flag, drawn from the mission files by reference. The portfolio does not replace the mission Viewboards; it reads them. It allows a dean, an incubator manager or a creative-economy unit to see at a glance which missions are on time, which have closed and what the campus has produced, and it is the natural source of aggregate creative-output data under the privacy rule.

6. Roles, Cadence and Rituals

6.1 Roles

IMF practice distinguishes the person in charge of a task (PIC) from the lead person in charge (LPIC). IMF-OS builds its role model on this distinction and adds the roles that creative-campus projects typically involve. Table 5 sets out the responsibilities of each role across the main activities of a mission.

Table 5. Roles in IMF-OS and their responsibilities across mission activities

Role	Open a version	Run the version	Close a version	Between versions
Lead owner (LPIC)	Writes purpose; declares Luaran; sets dates; announces recognition criteria	Keeps Viewboard accurate; checks evidence before counting	Enters done date; issues closing statement	Leads review; opens next version
Task owner (PIC)	Accepts assigned rows	Delivers tasks; links evidence; updates state	Confirms row states	Proposes carried-over and new tasks
Contributor	—	Helps with tasks	Acknowledged in closing	—
Mentor (lecturer)	Advises on feasibility of the declaration	Advises on quality	Signs off evidence at stage gates	Advises on next stage
Partner (hexa helix)	Consulted on requirements	Provides resources or evidence of use	Receives closing statement	Offers next opportunities
IMF steward	Provides template; checks essential fields	Supports teams	Updates portfolio Viewboard	Monthly portfolio review; applies privacy rule

The lead owner (LPIC) owns the mission. The LPIC opens each version by writing the purpose, declaring the outputs and setting the dates; keeps the Viewboard accurate; checks that evidence is linked before a task is counted; and closes the version with the closing statement. In student teams, the LPIC role is a leadership apprenticeship, and rotating it across versions spreads that experience.

The task owners (PICs) own rows. Each PIC delivers the tasks assigned to them, links the evidence and updates the state of their rows. A PIC may also propose extra tasks, which the LPIC records in the extra section.

Contributors are people who help with tasks without owning them, such as peers who test a prototype or volunteers at an event. Their help is acknowledged in the notes or in the closing statement.

The mentor, usually a lecturer, advises on quality and confirms that gates have been passed. The mentor does not run the mission; the role is closer to that of a reviewer who signs off evidence at the gates. Research on technology-startup incubation identifies guidance from experienced mentors as one of the strategies that help young ventures grow [39], which is the contribution the mentor role is meant to secure.

Partners from government, business, community, media and finance contribute resources, requirements and opportunities. They are consulted at the opening of versions, receive the closing statement and may provide evidence of use, such as a letter of interest or a showcase slot. Their visibility in the mission record is what makes multi-actor collaboration accountable without hierarchy [21, 26]. In practice we propose that partners receive a view-only link to the Viewboard summary, edit rights only for rows they own, for example a task to confirm a showcase slot or supply materials, and the closing statement at the end of each version. Partners who contribute creative material sign the ownership agreement of the idea stage. This gives partners enough visibility to trust the project and enough involvement to contribute evidence, without exposing individual achievement data.

The programme office or IMF steward maintains the portfolio Viewboard, provides templates and training, keeps the evidence folders in order and applies the privacy rule when records are aggregated.

Roles can be combined in small teams. In a team of two or three, the LPIC will normally also be a PIC and own rows of the Content grid, and every member is a PIC. Two combinations should be avoided. The mentor should not also be the LPIC, because the mentor's sign-off at gates would then lack independence, and the person who verifies evidence for a task should, where possible, not be the person who produced it. The steward role can be held by a mentor for a small cohort, provided that the privacy rule is applied to the mentor's own missions as to any other.

6.2 Cadence

The scheduler of IMF-OS runs on versions. We propose a default version length of two weeks, matching the sprint-like versions seen in practice, with monthly versions for slower work such as publication campaigns, which corresponds to the monthly cadence described in the 2020 paper. Within a version, the cadence has four moments: opening on day one, routine updates as tasks are completed, a mid-version check about halfway through and closing on the due date. Between versions, the team holds a short review and opens the next version. At portfolio level, the steward reviews all missions once a month. For a monthly version the same four moments apply: opening on day one, routine updates as tasks are completed, with at least one update a week, a mid-version check around day fifteen and closing on the due date. Figure 3 shows the cycle, drawn for the default two-week version.

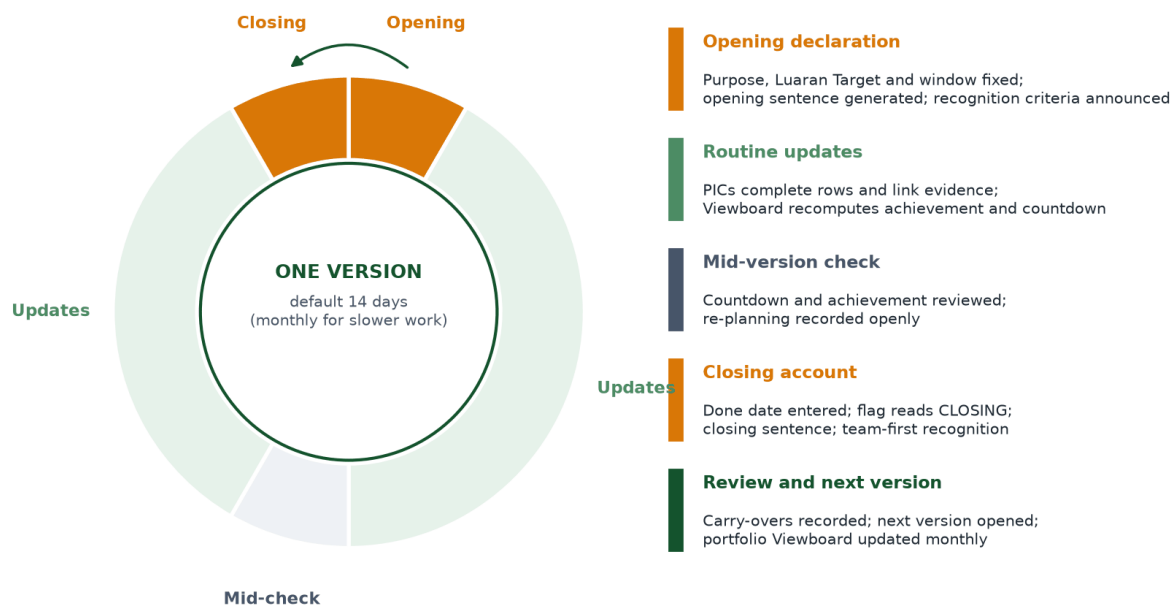


Figure 3. The IMF-OS version cycle: opening declaration, routine updates, mid-version check and closing account within a default two-week version, followed by review and the opening of the next version.

This cadence borrows from agile practice, which favours short iterations, frequent integration and adaptation; studies of artificial-intelligence product development show agile iteration as a response to the demand for rapid refinement and shorter time to market [42], and research on agile projects in Indonesia identifies the critical success factors and the cultural and managerial obstacles that such cadences must overcome [2]. It differs from typical agile practice in two ways. The version always ends with a closing flag and a narrative statement, which gives the cadence a clear emotional rhythm. And the version's outputs are counted against a declared target, which makes progress legible to people outside the team.

6.3 Rituals

Rituals are what turn a schedule into a shared experience. IMF practice contains two rituals that IMF-OS adopts and makes explicit.

The opening ritual is a declaration. In the practice sheets it takes the form of a formula-generated sentence beginning with “Bismillah” that states the IMF’s name, the number of outputs the team will strive to deliver, the purpose, the start and due dates and the number of days available. The sentence is computed from the Viewboard, so that it changes when the declaration changes. Its function is to commit the team publicly to a specific target in a specific window.

The closing ritual is an account. In practice it is a sentence beginning with “Alhamdulillah” that reports the percentage achieved, describes the result as a token of love from the team and states how many days early the work was completed. Its function is to close the loop: to state what was done, to thank those who did it and to mark the end of the version.

In the source institution these rituals are expressed in religious and affective language, and that language carries meaning for the teams who use it. IMF-OS treats the language as a contextual expression of a general function. A campus with a different culture, or a team with members of different faiths, can keep the function and change the words: an opening pledge and a closing thanks, a launch and a wrap, or phrases in the team’s own idiom. One multi-team IMF already mixes in greetings from another language, which shows that the ritual tolerates local variation. What matters is that the declaration is made before work begins and the account is given when the version closes. We state the principle explicitly: no member of a team should be required to use religious language, the wording of the rituals is chosen by the team, and recognition never depends on it. Inclusion also has a practical side. Not every student is comfortable with spreadsheets or owns a laptop, so the template should offer a simplified, mobile-friendly view in which a PIC can tick a task and paste an evidence link from a phone. Research on the spiritual capital of startup founders, which links spiritual resources to innovative work behaviour and well-being [43], and on human-centred technology in higher education [28], suggests why such rituals may matter, although neither tests IMF.

6.4 Recognition and Rewards

Recognition is the most delicate part of the shell. Practice shows several forms: most-valuable-contributor cells for lead and task owners, rankings by achievement, statements of how many days early a version was finished and, in one gamified IMF, leagues and medal counts. The founding texts mention none of these as rewards, although the blog presents the ability to see the best and the lowest contributor as a benefit of the method.

The literature gives reasons both to use recognition and to use it carefully. Gamification with rewards, challenges and points can raise engagement and growth in digital startups [18], and gamified learning systems have been found to raise student motivation and performance [44]. Rewards and autonomy act as job resources that support work engagement [45], incentives and motivation are associated with productivity [46], and achievement motivation is linked with innovative work behaviour [47]. Studies of workplace happiness among young startup employees highlight the importance of autonomy and organisational justice for retention [48]. On the other hand, performance parameters can create stress that harms the work they are meant to improve [5].

IMF-OS therefore adopts four safeguards. Recognition celebrates; it does not sanction, and achievement data are not used for formal evaluation without the team’s agreement. Recognition is team-first: the closing statement celebrates the mission before any individual. Criteria are announced at opening, so that nobody is surprised by what counts. And only verified, planned work counts towards achievement-based recognition, so that splitting tasks or adding easy extras cannot inflate a ranking. Material rewards, where a campus chooses to offer them, should follow the same rules and should be stated in the opening declaration.

Table 6 summarises the cadence, rituals and recognition with their purposes and safeguards.

Table 6. Cadence, rituals and recognition in IMF-OS

Moment	When	IMF element	Purpose	Safeguard
Opening declaration	Day 1 of a version	Opening sentence computed from Viewboard	Public commitment to target and window	Targets fixed at opening; later tasks recorded as extra
Routine updates	As tasks complete	Content rows, evidence links	Keep the record live	A task counts only with evidence
Mid-version check	About halfway	Countdown and achievement	Re-plan early if behind	Re-planning recorded, not hidden
Closing account	Due date	Done date, closing flag, closing sentence	Close the loop; thank contributors	Close whatever the achievement
Recognition	At closing	MVP cells, team celebration, early-finish line	Sustain voluntary effort [18,45]	Celebrate, not sanction; team first; criteria declared; verified planned work only
Portfolio review	Monthly	Portfolio Viewboard	Campus overview and support	Aggregate fields only

7. An Illustrative Mission Template

7.1 Purpose and Status of the Template

This section presents a worked mission template to show how IMF-OS would look in use. The template is illustrative. The project, the team, the dates and all values are invented for this paper; they do not describe a real project, and no real person’s data are used. Role labels replace names. The template follows the field structure of the practice sheets described in Section 2.4, with the rules of Section 4.4 applied.

7.2 The Scenario

Imagine a team of four students, a lecturer mentor and a partner from a local textile cooperative. They want to show an augmented-reality app that animates traditional textile motifs during the week of the World Conference on Creative Economy in Jakarta. The scenario uses the conference only as an example of an external deadline; it implies no affiliation with, or endorsement by, the conference or its organisers. Their idea stage has already closed. The template shows the first prototype-stage version, which runs for fourteen days and must close the day before the conference opens. Figure 4 shows the Viewboard and five selected rows of the Content grid as they might appear on day ten of the version; Table 7 lists all nine rows.

7.3 The Viewboard

The Viewboard of the illustrative mission is titled “IMF Wastra AR Showcase v1.1”. Its purpose statement reads: “To build and test a working AR prototype of three textile motifs and prepare it for a public showcase.” Luaran Target is eight, the number of planned tasks; Luaran Extra is one, a short paper draft that a PIC proposed after opening; Luaran Total is therefore nine. The start date is 7 October 2026 and the due date is 20 October 2026. The countdown formula returns the number of days left, the closing flag reads “NOT CLOSING” until a done date is entered, and planned achievement is computed as verified planned tasks divided by Luaran Target. On day ten, in the illustration, five of the eight planned tasks are verified, so planned achievement is 62.5 percent, and the extra paper draft is in progress.

The opening sentence, rendered in English and following the pattern of the practice sheets, reads: “Bismillah. With IMF Wastra AR Showcase v1.1, our team strives to deliver 8 outputs,

Viewboard IMF Wastra AR Showcase v1.1				ILLUSTRATIVE: Invented values	
Tujuan (purpose) Build and test a working AR prototype of three textile motifs and prepare it for a public showcase		Luaran Target 8	Luaran Extra 1	Luaran Total 9	Closing NOT CLOSING
Start 7 Oct 2026	Due 20 Oct 2026	Countdown 4 Hari Lagi	Capaian (planned) 62.5%	Verified / Target 5 / 8	Mission owner LPIC (role label)
<i>"Bismillah. With IMF Wastra AR Showcase v1.1, our team strives to deliver 8 outputs ... We begin on 7 October 2026 and finish on 20 October 2026, in 14 days." (opening sentence computed from the Viewboard; wording adaptable)</i>					
Code	Task	Owner	Checklist	Capaian	Evidence
WASTRA #01	Digitise three textile motifs as vector files	PIC-1	☒	100%	Disini
WASTRA #02	Build AR marker detection for the three motifs	PIC-2	☒	100%	Disini
WASTRA #05	Run user test with ten visitors	PIC-4	☒	100%	Disini
WASTRA #06	Revise build after user test	PIC-2	☐	0%	(pending)
WASTRA #09	EXTRA: draft short paper on the prototype	PIC-4	☐	0%	(pending)
<i>Content grid (first rows shown). Capaian (planned) = verified planned tasks / Luaran Target; extra tasks counted separately.</i>					

Figure 4. Illustrative Viewboard and Content grid of the invented mission “IMF Wastra AR Showcase v1.1” on day ten of a fourteen-day version. All values are fictional; role labels replace names. Five selected Content rows are shown; Table 7 lists all nine.

with the purpose of building and testing a working AR prototype of three textile motifs. We begin on 7 October 2026 and finish on 20 October 2026, in 14 days.” A team that prefers secular language would write “We commit to ...” in place of the first word. The closing sentence, which the formula would produce only after the done date is entered, follows the pattern: “Alhamdulillah. This is the token of love from our team to deliver the outputs of IMF Wastra AR Showcase v1.1. Of the 100 percent target over 14 days, we achieved [achievement] percent, finishing [days] days ahead of schedule.”

7.4 The Content Grid

Table 7 lists the Content rows of the illustrative mission. Each task carries a code formed from the mission name and a number, a description, an owner role, a state and the kind of evidence that would be linked. Tasks one to eight are planned; task nine is extra. The table shows how the evidence for each task doubles as material for later stages: the provenance file supports a future copyright recording, the user-test record supports the pitch deck, and the showcase slot confirmation is evidence of the partner’s involvement.

Table 7. Content rows of the illustrative mission “IMF Wastra AR Showcase v1.1” (invented values; role labels instead of names)

Code	Task	Owner	State on day 10	Evidence to be linked
WASTRA #01	Digitise three textile motifs as vector files	PIC-1	Verified	Vector files in shared folder
WASTRA #02	Build AR marker detection for the three motifs	PIC-2	Verified	Build link and screen recording
WASTRA #03	Animate the three motifs	PIC-1	Verified	Animation files

Code	Task	Owner	State on day 10	Evidence to be linked
WASTRA #04	Assemble provenance file of design drafts	PIC-3	Verified	Dated drafts and authorship notes
WASTRA #05	Run user test with ten visitors	PIC-4	Verified	Test record
WASTRA #06	Revise build after user test	PIC-2	In progress	Revised build link
WASTRA #07	Confirm showcase slot with partner	LPIC	In progress	Partner confirmation
WASTRA #08	Prepare one-page showcase brief	PIC-3	Open	Brief file
WASTRA #09 (extra)	Draft short paper on the prototype	PIC-4	In progress	Draft file

7.5 Closing and Chaining

On 20 October 2026, in the illustration, the LPIC enters the done date, the closing flag changes to “CLOSING”, and the closing sentence is generated. Whatever the final achievement, the version is closed. The team holds a short review, records any unfinished planned task as carried over, and opens the next version, perhaps “IMF Wastra AR Showcase v1.2”, with a new declaration. If the showcase succeeds, the project may move to the startup track, with an incubator application and a pitch deck as Luaran, and to the paper track, with the extra draft promoted to a planned output. The mission chain thus records the project’s history version by version, and the portfolio Viewboard shows the campus that one more creative-campus project has produced a tested prototype and a public showcase.

8. Analytical Evaluation

8.1 Walkthrough Against the Requirements

We evaluate IMF-OS first by walking through the six requirements of Section 3.3 and asking which elements of the framework address each one. The walkthrough is analytical: it shows how the framework is designed to meet each requirement, not that it does so in practice.

R1, countable and evidenced outputs, is addressed by the declaration rule, the evidence rule and the Luaran fields. Because outputs are counted from coded tasks and each task requires a link to evidence, the mission record states not only how much was done but where the proof is. For creative-campus projects this matters twice: once for the team’s own control, and again when a prototype, a design or a recording becomes the basis for an intellectual-property application, an incubator review or a paper.

R2, multi-actor accountability without hierarchy, is addressed by the role model and the Content grid. Every row has an owner, the lead owner is named, and partners appear in the mission record as consulted parties and providers of evidence. Accountability rests on visibility rather than on reporting lines, which suits teams whose members belong to different organisations.

R3, alignment with external deadlines, is addressed by the scheduler. The due date of a version can be set to the day before an external deadline, the countdown makes the remaining time visible to everyone, and the closing rule ensures that the version ends on time with an honest record. The illustrative mission in Section 7 shows a version timed to close the day before a major conference.

R4, low cost and a low skill threshold, is addressed by the choice of medium. IMF-OS runs on a shared spreadsheet with a handful of formulas: counts, sums, a date difference and two conditional statements. The practice sheets show that the method has been used for years across many kinds of work with no specialised software, which is the strongest available evidence of feasibility.

R5, inclusive and sustainable motivation, is addressed by the shell. The opening and closing rituals give each version meaning; recognition celebrates achievement; and the four safeguards limit the risks that metrics create. The separation of the ritual’s function from its religious expression

allows teams of different backgrounds to share the method.

R6, aggregatable and protected records, is addressed by the two-number rule, the portfolio Viewboard and the privacy rule. Because every mission reports planned achievement and extra output in the same way, missions can be compared and combined; because aggregation removes or pseudonymises contributors, the combined records can be shared without exposing individuals. Table 8 summarises the walkthrough.

Table 8. Analytical walkthrough of IMF-OS against the design requirements

Requirement	IMF-OS elements that address it	How it is addressed	What a pilot should check
R1 Countable, evidenced outputs	Declaration rule; evidence rule; Luaran fields	Outputs counted from coded tasks with evidence links	Share of counted tasks with valid evidence
R2 Accountability without hierarchy	Role model; Content owners; partners in record	Visibility of ownership across organisations	Share of rows with a named owner; partner involvement
R3 External deadlines	Scheduler; countdown; closing rule	Version due dates set before external deadlines	Share of versions closed on the due date
R4 Low cost, low skill	Shared spreadsheet; few formulas; template	No specialised software; short training	Training time; template errors
R5 Inclusive motivation	Rituals; recognition; safeguards	Meaning and celebration with limits on metric pressure	Active contributors across versions; perceived fairness
R6 Aggregatable, protected records	Two-number rule; portfolio; privacy rule	Comparable mission records shared without personal data	Agreement of recomputed figures; privacy compliance

8.2 Comparison with Related Methods

We evaluate IMF-OS second by comparing it with related methods that a campus might use instead. The comparison is not a contest. Several of these methods can run inside IMF-OS, and IMF-OS can borrow from all of them.

Agile methods share IMF-OS’s short, time-boxed iterations and its emphasis on visible progress. Research on agile projects in Indonesia identifies critical success factors and highlights cultural constraints, limited training, unclear guidelines and insufficient management support as obstacles [2], and work on artificial-intelligence products shows agile iteration as a way of shortening time to market [42]. IMF-OS adds a declared output count, a single achievement figure and opening and closing rituals, and it asks for less training than a full agile framework. A team that uses agile ceremonies can record its sprints as IMF versions.

Earned-value management shares IMF-OS’s concern with comparing work done against work planned. Applied to construction projects, it combines cost and schedule indicators to show whether a project is ahead or behind [15]. IMF-OS uses a simpler measure, verified outputs over declared outputs, because creative-campus projects rarely have reliable cost baselines. Where a project does have a budget, earned-value indicators can be added to the Viewboard.

Project-management maturity models assess how well an organisation manages projects and recommend improvements in a directed way [1]. They operate at the level of the organisation rather than the mission. IMF-OS is a mission-level routine, but its portfolio view can supply the evidence that a maturity assessment needs, and a companion paper in this series proposes an indicator set and a maturity model built on IMF records [9].

Task-management systems and monitoring dashboards integrate the recording, assignment and reporting of tasks and display performance indicators [12, 13]. They are close to IMF’s Content

grid and Viewboard in function. What IMF-OS adds is the mission boundary, with a declared target and a closing rule, and the shell of rituals and recognition. Purpose-built monitoring platforms for grants show what a software version of IMF-OS could offer in real-time tracking and structured reporting [3, 4]. Visual task boards of the Kanban type, with columns such as to do, doing and done, are already familiar to many student teams. They make the flow of work visible and limit work in progress, but they usually have no declared target, no achievement ratio and no closing date. A team that prefers a board can keep it as its working view and treat the IMF Content grid as the record into which finished, evidenced cards are entered.

Design thinking and lean startup describe how to move from a problem to a tested solution [6, 32]. They supply the content of the idea and prototype stages; IMF-OS supplies the container. A team that follows design thinking can declare empathy interviews, problem definitions, concepts and prototypes as Luaran, and a team that follows lean startup can declare hypotheses, experiments and learning records. Project-based learning models can be run in the same way [20]. Table 9 summarises the comparison.

Table 9. IMF-OS compared with related methods

Method	Shares with IMF-OS	IMF-OS adds	How to combine
Agile iterations [2, 42]	Short time-boxed cycles; visible progress	Declared output count; achievement figure; rituals; lower training need	Record sprints as IMF versions
Earned-value management [15]	Work done against work planned	Output-based measure without cost baseline	Add cost indicators where budgets exist
Maturity models [1]	Concern with improving project management	Mission-level routine and records	Use portfolio records as maturity evidence
Task systems and dashboards [12, 13]	Task recording, assignment, status display	Mission boundary; closing rule; shell	Use as software front end for IMF records
Design thinking and lean startup [6, 32]	Iterative path from problem to tested solution	Container with targets, owners and closure	Declare their steps as Luaran
Kanban-type task boards	Visible flow of work; limits on work in progress	Declared target; achievement ratio; closing date	Keep the board as working view; enter finished, evidenced cards in the Content grid

8.3 Traceability: What Is Documented, Observed and Proposed

Table 10 traces each element of IMF-OS to its source. Elements are labelled documented when they appear in the founding texts, observed when they appear in the practice sheets but not in those texts, and proposed when they are introduced in this paper. The mission, the Viewboard, the Content grid with coded tasks and the monthly version are documented. The countdown, closing flag, Luaran Target and Extra, lead and task owners, evidence links, opening and closing sentences and most-valuable-contributor cells are observed. The two-number rule, the privacy procedure, the mid-version check, the gate decisions, mission chaining, the portfolio Viewboard, the mentor and steward roles, the recognition safeguards and the separation of ritual function from expression are proposed. The table shows that most of the operating machinery of IMF-OS already exists in practice; what the framework adds is mainly the rules, the lifecycle and the governance that let that machinery run across many projects.

Table 10. Traceability of IMF-OS elements: documented in the founding texts, observed in practice or proposed in this paper

Element	Source	Status
Mission with objectives, outputs and contributors	Blog 2019; paper 2020	Documented
Viewboard and Content grid; unique task codes	Paper 2020	Documented
Monthly versions	Paper 2020	Documented
Countdown, done date, closing flag	Practice sheets	Observed
Luaran Target, Extra and Total; achievement percentage	Practice sheets	Observed
Lead and task owners (LPIC, PIC); MVP cells	Practice sheets	Observed
Evidence links; planned and extra sections	Practice sheets	Observed
Opening and closing sentences	Practice sheets	Observed
Two-week sprint versions; multi-team roll-ups	Practice sheets	Observed
Five layers and the operating-system reading	This paper	Proposed
Declaration and evidence rules	Practice made explicit	Proposed (formalised)
Two-number rule; closing rule	This paper, from practice variation	Proposed
Privacy rule and procedure	This paper; Law No. 27 of 2022	Proposed
Mission chaining, stage gates and gate decisions	This paper	Proposed
Portfolio Viewboard	This paper, generalising roll-ups	Proposed
Mentor, partner and steward roles; role combinations	This paper	Proposed
Mid-version check; monthly portfolio review	This paper	Proposed
Separation of ritual function from expression; recognition safeguards	This paper	Proposed
Ownership agreement; mobile view	This paper	Proposed

8.4 Why IMF-OS Is an Essential Tool for Creative-Campus Projects

The walkthrough and the comparison support a specific claim. A creative-campus project must do six things: declare what it will produce, divide the work among people who do not report to one another, document what has been done, keep to deadlines set by others, keep people motivated and leave records that the campus can use. Some method must perform these functions, whether formally or informally. IMF-OS performs all six with one routine, on tools that every campus already has, in a form that has been used for years in the institution that created it. The other methods perform some of the functions, often better in their own domain, but none performs all six with so little overhead.

We therefore call IMF-OS essential in a functional sense. It is not the only possible operating system, and a campus could build an equivalent from other parts. But the functions it performs are indispensable for creative-campus projects, its fit with those functions is close, and its cost is low. Whether it also improves outcomes, such as the share of projects that close on time or the number of verified creative outputs per project, is an empirical question that this paper does not answer. We can, however, state what evidence would count against the claim. If, in a pilot, teams using IMF-OS closed fewer versions on time than comparable teams without it, produced no more verified outputs, reported the routine as a burden that outweighed its benefits, or found the records unusable for aggregation, the claim of essentiality would have to be revised or abandoned. The indicators in the last column of Table 8 are designed to detect exactly these outcomes.

9. Discussion

9.1 Contributions

The paper contributes to the understanding of IMF in three ways. It reviews the descriptions and practice of IMF from the perspective of use, showing that practice has moved well beyond the founding texts by adding bounded missions, differentiated roles, evidence links, rituals and recognition. It proposes a new operational idea of IMF as an operating system, with five layers, five rules, a lifecycle of chained missions, a role model, a cadence, rituals with safeguarded recognition and a portfolio view. And it provides an illustrative template that shows how the system would look in a creative-campus project.

It also contributes to the discussion of how universities can take part in the creative economy. Policy documents name academia as one of the helices of the creative economy and stress talent, intellectual property and data. They say less about the everyday routines through which a campus turns ideas into outputs. IMF-OS is one answer at the level of routine: a way of running projects that produces the documented, attributable outputs that the legal framework presumes and the aggregate records that the master plan asks for.

9.2 Guidance for Adoption

A campus that wants to adopt IMF-OS can start small. We suggest five steps. First, appoint an IMF steward and prepare a template with the essential Viewboard fields, the Content columns and the formulas. Second, choose three to five creative-campus projects for a first cohort and train their lead owners in an afternoon session. Third, run two versions with the default two-week cadence, with the steward attending the openings and closings. Fourth, build the portfolio Viewboard after the first versions close, using only aggregate fields. Fifth, review the experience with the teams and adjust the template, the cadence and the language of the rituals before extending the method to more projects. Programmes that build creative-entrepreneurship capacity through training and mentoring [27, 33] provide natural settings for such a cohort.

9.3 Risks

Three risks deserve attention. The first is metric gaming. A single achievement figure invites behaviour that moves the number rather than the work: splitting tasks, declaring easy outputs or ticking boxes without evidence. The declaration rule, the evidence rule and the two-number rule are designed to limit this, but stewards and mentors must still look at the evidence and not only at the percentage.

The second is surveillance. A Viewboard that names contributors and shows their achievement can feel like a monitoring device rather than a shared tool, especially if it is used for formal evaluation. The recognition safeguards and the privacy rule are designed to prevent this, and the human-centred approach to technology in higher education offers a further reason to keep the system in the team's hands [28].

The third is exclusion. Rituals expressed in the language of one faith or one culture may make others feel like guests. IMF-OS separates the function of the rituals from their expression so that each team can choose its own words, and we recommend that campuses make this choice explicit when they introduce the method.

9.4 Limitations

The framework has four main limitations. It is a design proposal that has not been piloted in the form presented here; the practice sheets show that its components work in the source institution, but they do not show that the combined system works elsewhere. The mapping of POAC onto IMF's components is our inference from practice and has not been confirmed by the method's originator. Our evidence about practice comes from the structure of a limited number of sheets examined in depth and others seen only in part; other IMFs may differ. And the illustrative

template, being invented, cannot reveal the difficulties that a real team would encounter.

9.5 Future Work

Three lines of work follow. The first is a pilot: a cohort of creative-campus projects run on IMF-OS for at least two versions, with records of closing, achievement and team experience. The second is measurement: an indicator set and maturity model that would allow campuses to assess how well IMF is used and what it produces, which is the subject of the companion measurement paper in this series [9]. The third is tooling: a lightweight software version of IMF-OS that keeps the spreadsheet’s simplicity while adding the portfolio view, evidence storage and privacy controls, building on existing monitoring platforms [3]. Such tooling will increasingly include AI assistants that can draft task lists, update states from linked files and summarise progress. We suggest a simple boundary: assistants may propose and update, but declarations, evidence verification, gate decisions and recognition remain human, because these are the points at which a team decides what its work means and whether it is finished.

10. Conclusion

Creative-campus projects need a way of turning ideas into outputs that is simple enough for students, rigorous enough for partners and motivating enough for volunteers. This paper has argued that IMF, a spreadsheet-based method created at an Indonesian university in 2019, can serve as that way when it is understood as an operating system.

We reviewed how IMF has been described. The founding blog post and conference paper present it as a low-cost means of implementing POAC through a Viewboard and a Content grid. The citing literature mostly mentions it without engaging with it. Practice, by contrast, has turned it into a rich system of bounded missions, coded and evidenced tasks, lead and task owners, opening and closing rituals and recognition.

On this basis we proposed a new operational idea of IMF: IMF-OS, the operating system of a creative-campus project, in which a lead owner opens each version with a declaration on the Viewboard, task owners deliver and evidence coded tasks in the Content grid, the Viewboard computes achievement and time remaining, and the team closes the version with a public account before the next stage begins. The framework has five layers, five rules, a lifecycle of chained missions leading to a creative startup, a paper or an event, a role model, a cadence and a portfolio view, and we illustrated it with an invented mission timed for a major creative-economy event.

The analytical evaluation suggests that IMF-OS meets the six requirements of creative-campus projects and complements rather than replaces agile, earned-value, design-thinking and lean-startup practice. We call it essential because the functions it performs are indispensable for such projects and because it performs them all at low cost. Whether it improves outcomes is the question that pilots and the measurement instrument proposed in the next paper must answer.

Appendix A. Glossary of IMF Terms

Table 11 explains the Indonesian and IMF-specific terms used in this paper.

Table 11. Glossary of IMF terms used in this paper

Term	Literal meaning	Function in IMF-OS
Tujuan	Objective, purpose	Purpose statement on the Viewboard
Luaran (Target, Extra, Total)	Output	Count of planned outputs, outputs beyond the plan, and their sum
Capaian	Achievement	Verified planned tasks divided by Luaran Target

Term	Literal meaning	Function in IMF-OS
Countdown; Hari Lagi	Days remaining	Formula showing days left until the due date
CLOSING / NOT CLOSING	Closing flag	Shows whether a version has been closed
Lovecode (Code)	Love code	Unique code of a task, formed from mission name and number
Disini	Here	Label of the evidence link in a task row
PIC; LPIC	Person in charge; lead person in charge	Task owner; mission owner
MVP LPIC; MVP PIC	Most valuable lead and task owner	Recognition cells
Bismillah; Alhamdulillah	In the name of God; praise be to God	Opening and closing words of the ritual sentences in the source institution; adaptable
Tanda cinta	Token of love	Phrase in closing sentences describing the team's achievement
Viewboard	View board	Mission record: header of an IMF
Extra Job; To Do	Extra work; planned work	Sections of the Content grid

Declarations

Funding. This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Conflict of interest. The authors declare no conflict of interest. The originator of the IMF method is cited in this paper and is not an author.

Data availability. No new data involving human participants were collected. The study used published documents and the structure of spreadsheets accessed read-only; no personal data were extracted. The mission template in Section 7 is invented for illustration and contains no real data.

Use of AI tools. Generative AI tools were used to assist with language editing, drafting support and formatting of this manuscript. The authors reviewed and verified all content, quotations and references and take full responsibility for the work.

References

- [1] I. 'ilma Insyifani, T. Raharjo, E. K. Budiardjo, and K. Mahatma, "Project Management Maturity Improvement based on KPMMM Framework: A Case Study of an IT Consultant PQR," *Technomedia J.*, vol. 8, no. 1, pp. 57–71, May 2023, doi: [10.33050/tmj.v8i1SP.2005](https://doi.org/10.33050/tmj.v8i1SP.2005).
- [2] T. Raharjo, "A Model of Critical Success Factors for Agile Information Technology Project in Indonesia using Analytic Hierarchy Process (AHP)," *ADI J. Recent Innov.*, vol. 5, no. 1, pp. 68–77, Aug. 2023, doi: [10.34306/ajri.v5i1Sp.968](https://doi.org/10.34306/ajri.v5i1Sp.968).
- [3] U. Rahardja, L. Sulisty, D. Safarina, M. Rapidan Kusuma, N. Silawati, and Z. Nanle, "HibahQu Education Monitoring Platform Based on Human-Centric Orange Technology Laravel 12 Vue.js," *ADI Bisnis Digit. Interdisiplin J.*, vol. 6, no. 2, pp. 203–218, Dec. 2025, doi: [10.34306/abdi.v6i2.1348](https://doi.org/10.34306/abdi.v6i2.1348).
- [4] I. P. Gustiah, N. Lutfiani, M. R. Kusuma, E. P. Lestari, and T. Green, "Grant Submission Monitoring System Based on Orange Technology Using Laravel 12 and Vue.js," *J. Orange Technol.*, vol. 1, no. 2, pp. 75–90, Apr. 2025, doi: [10.68012/jot.v1i2.24](https://doi.org/10.68012/jot.v1i2.24).
- [5] A. S. Bist, B. Rawat, U. Rahardja, Q. Aini, and A. G. Prawiyogi, "An Exhaustive Analysis of Stress on Faculty Members Engaged in Higher Education," *IAIC Trans. Sustain. Digit. Innov.*, vol. 3, no. 2, pp. 126–135, Apr. 2022, doi: [10.34306/itsdi.v3i2.533](https://doi.org/10.34306/itsdi.v3i2.533).
- [6] D. Abbas, K. Siahaan, and M. Yusup, "Design Thinking as a Business Model for Empowering Creative Entrepreneurs in the Digital Era," *Startuppreneur Bus. Digit.*, vol. 4, no. 2, pp. 124–133, Oct. 2025, doi: [10.33050/sabda.v4i2.805](https://doi.org/10.33050/sabda.v4i2.805).

- [7] F. Zidan, D. Nugroho, R. Asri, and S. Agustina, "Peran Pendidikan dalam Mempersiapkan Generasi Creativepreneur di Masa Depan," *ADI Bisnis Digit. Interdisiplin J.*, vol. 5, no. 1, pp. 41–46, Jun. 2024, doi: [10.34306/abdi.v5i1.1099](https://doi.org/10.34306/abdi.v5i1.1099).
- [8] N. Lachlan, R. F. Terizla, and A. Lansonja, "Rethinking IMF: An Evolutionary Concept Analysis and Redefinition of Indicator Measurement Factor Analysis for Creative-Campus Projects," *CreatiCampus Working Paper Series*, vol. 1, no. 1, Esquine CreatiResearch Institute, Oct. 2026. [Online]. Available: <https://creaticampus.esquine.com/papers/rethinking-imf-indicator-measurement-factor-analysis-creative-campus/>
- [9] N. Ani, A. Erica, and B. Callula, "Measuring What IMF Manages: A Design-Science Protocol for an Indicator Set and Maturity Model for CreatiCampus Projects," *CreatiCampus Working Paper Series*, vol. 1, no. 1, Esquine CreatiResearch Institute, Oct. 2026. [Online]. Available: <https://creaticampus.esquine.com/papers/measuring-what-imf-manages-indicator-set-maturity-model-creative-campus/>
- [10] N. Lutfiani, U. Rahardja, and K. T. Khasanah, "The Development Viewboard As an Information Media at Official Site Asosiation," *Aptisi Trans. Manage.*, vol. 6, no. 1, pp. 10–18, Jan. 2022, doi: [10.33050/atm.v6i1.1529](https://doi.org/10.33050/atm.v6i1.1529).
- [11] U. Rahardja, "Meningkatkan Kualitas Sumber Daya Manusia Dengan Sistem Pengembangan Fundamental Agile," *ADI Bisnis Digit. Interdisiplin J.*, vol. 3, no. 1, pp. 63–68, Jun. 2022, doi: [10.34306/abdi.v3i1.760](https://doi.org/10.34306/abdi.v3i1.760).
- [12] N. P. Sulistyadewi, G. Sastrawangsa, and R. Wulandari, "Task Management System Design at CV. Balinet Intermedia Using MVC (Model-View-Controller)," *Technomedia J.*, vol. 8, no. 2, pp. 154–169, Apr. 2023, doi: [10.33050/tmj.v8i2.1979](https://doi.org/10.33050/tmj.v8i2.1979).
- [13] S. D. Sugiyanti, R. Widayanti, M. B. Ulum, G. Firmansyah, and A. H. Azizah, "Design Dashboard Monitoring Teacher Performance Assessment at Cinta Kasih Tzu Chi High School," *IAIC Trans. Sustain. Digit. Innov.*, vol. 4, no. 1, pp. 46–56, Oct. 2022, doi: [10.34306/itsdi.v4i1.569](https://doi.org/10.34306/itsdi.v4i1.569).
- [14] Z. Sekar Puranti, W. Yuwono, and R. Asmara, "Monitoring Proyek Akhir Mahasiswa Berbasis Android Pada Sistem Informasi Manajemen PENS," *Technomedia J.*, vol. 6, no. 2, pp. 138–151, 2022, doi: [10.33050/tmj.v6i2.1702](https://doi.org/10.33050/tmj.v6i2.1702).
- [15] S. Suriyanto, A. Purkon, W. Aswawarman, T. S. Soeparyanto, and A. Kadir, "Performance Evaluation of Costs and Schedules for the Merah Putih Cooperative Construction Project Using Earned Value Management," *ADI Bisnis Digit. Interdisiplin J.*, vol. 7, no. 1, pp. 105–116, Jun. 2026, doi: [10.34306/abdi.v7i1.1483](https://doi.org/10.34306/abdi.v7i1.1483).
- [16] A. Nuche, O. Sy, and J. Carlos Rodriguez, "Optimizing Efficiency Through Sustainable Strategies: The Role of Management and Monitoring in Achieving Goals," *Aptisi Trans. Manage.*, vol. 8, no. 2, pp. 167–174, May 2024, doi: [10.33050/atm.v8i2.2257](https://doi.org/10.33050/atm.v8i2.2257).
- [17] L. Larisang, S. Sanusi, M. A. Bora, and A. Hamid, "Practicality and Effectiveness of New Technopreneurship Incubator Model in The Digitalization Era," *Aptisi Trans. Technopreneurship*, vol. 7, no. 2, pp. 318–333, Jul. 2025, doi: [10.34306/att.v7i2.482](https://doi.org/10.34306/att.v7i2.482).
- [18] F. Rahardja, L. K. Choi, R. C. Wijaya, and R. A. Sunarjo, "Gamification in Digital Startups: Enhancing User Engagement and Business Growth," *Startuppreneur Bus. Digit.*, vol. 4, no. 1, pp. 1–11, Apr. 2025, doi: [10.33050/sabda.v4i1.639](https://doi.org/10.33050/sabda.v4i1.639).
- [19] S. Yunita, A. Gandamana, W. Lubis, F. Rachman, and S. Bali, "Innovative Learning Strategies by Embedding Design Thinking-based Project Learning in Textbooks for Edupreneurial Impact," *Aptisi Trans. Technopreneurship*, vol. 7, no. 2, Jul. 2025, doi: [10.34306/att.v7i2.539](https://doi.org/10.34306/att.v7i2.539).
- [20] T. Suminar, J. Sutarto, Y. Siswanto, and A. D. Cahyani, "Creating a Technopreneurship-Based Life Skills Education Model with Methods of Project-Based Learning," *Aptisi Trans. Technopreneurship*, vol. 7, no. 2, pp. 567–581, Jul. 2025, doi: [10.34306/att.v7i2.576](https://doi.org/10.34306/att.v7i2.576).
- [21] M. H. R. Chakim, U. Rahardja, E. D. Astuti, E. Erika, and C. T. Hua, "The Social Empowerment Role of the Penta Helix Entrepreneurship Ecosystem in Driving the National Economy," *ADI Pengabd. Masy. J.*, vol. 6, no. 1, pp. 1–13, Sep. 2025, doi: [10.34306/adimas.v6i1.1283](https://doi.org/10.34306/adimas.v6i1.1283).

- [22] G. S. Waskita, D. R. Widi Ashari, S. Izzah, D. Rahmawati, and A. Permadi, “Pentahelix Collaboration for MSME Entrepreneurship Ecosystem Development during National Santri Day,” *ADI Pengabd. Masy. J.*, vol. 6, no. 2, pp. 224–237, Jun. 2026, doi: [10.34306/adimas.v6i2.1463](https://doi.org/10.34306/adimas.v6i2.1463).
- [23] G. S. Waskita, D. R. W. Ashari, M. B. Al Haris, A. Rifa'i, and S. Izzah, “Value Co-Creation in The Gig Economy: Phenomenological Study of Micro Entrepreneurs in Emerging Indonesian Creative Hubs,” *ADI Bisnis Digit. Interdisiplin J.*, vol. 7, no. 1, pp. 58–69, Jun. 2026, doi: [10.34306/54g0he22](https://doi.org/10.34306/54g0he22).
- [24] U. Rahardja, N. Ani, and N. Leonie, “‘Inclusively Creative’ Campuses: An Orange Technology Framework for Humanistic Creative-Economy Higher Education in Indonesia,” *CreatiCampus Working Paper Series*, vol. 1, no. 1, Esquine CreatiResearch Institute, Oct. 2026. [Online]. Available: <https://creaticampus.esquine.com/papers/inclusively-creative-campus-orange-technology-wcce-2026/>
- [25] U. Rahardja, M. Budiarto, K. Lutfiyah, O. F. P. Wahyudi, I. K. H. Azz, N. Azizah, and D. Julianingsih, “Analysis of the Effectiveness of Visual Language and Narrative in Conveying Value Propositions in Pitching Decks,” *Int. Trans. Artif. Intell.*, vol. 3, no. 2, pp. 161–170, May 2025, doi: [10.33050/italic.v3i2.807](https://doi.org/10.33050/italic.v3i2.807).
- [26] S. Wijono, U. Rahardja, H. D. Purnomo, N. Lutfiani, and N. A. Yusuf, “Leveraging Machine Learning Models to Enhance Startup Collaboration and Drive Technopreneurship,” *Aptisi Trans. Technopreneurship*, vol. 6, no. 3, Nov. 2024, doi: [10.34306/att.v6i3.462](https://doi.org/10.34306/att.v6i3.462).
- [27] N. F. Hermiati, M. Muhtarom, M. F. Hilmi, L. Pasha, and L. Devi, “Reframing Community Empowerment through Creativepreneurship as a Multidisciplinary Social Service Model,” *ADI Pengabd. Masy. J.*, vol. 6, no. 2, pp. 173–185, May 2026, doi: [10.34306/adimas.v6i2.1472](https://doi.org/10.34306/adimas.v6i2.1472).
- [28] S. Wibowo, I. A. Widjaya, J. Zanubiya, R. Evans, and U. Rahardja, “Orange Technology for Humanistic Innovation in Higher Education,” *J. MENTARI*, vol. 4, no. 2, pp. 105–115, Mar. 2026, doi: [10.33050/mentari.v4i2.899](https://doi.org/10.33050/mentari.v4i2.899).
- [29] N. Leonie, N. Lachlan, R. F. Terizla, and A. Lansonnia, “Links That Work, Bets That Stack: A Critical Case Study of the ‘Muskonomy’ as an Interlinked Venture Ecosystem and Its Lessons for Indonesian Creative Campuses,” *CreatiCampus Working Paper Series*, vol. 1, no. 1, Esquine CreatiResearch Institute, Oct. 2026. [Online]. Available: <https://creaticampus.esquine.com/papers/muskonomy-interlinked-venture-ecosystem-indonesian-creative-campuses/>
- [30] E. A. Beldiq, G. Jacqueline, A. Husain, and N. Ani, “Pricing the Bet Stack: A Segment-Level Monte Carlo and Real-Options Valuation of SpaceX to 2030 and Its Lessons for Indonesian Retail and Student Investors,” *CreatiCampus Working Paper Series*, vol. 1, no. 1, Esquine CreatiResearch Institute, Oct. 2026. [Online]. Available: <https://creaticampus.esquine.com/papers/pricing-the-bet-stack-spacex-valuation-indonesian-investors/>
- [31] J. Reinaldy Mansa, S. Anggun Pratama, W. Wirdayanti, and D. S. Angreni, “Optimizing User Interface of MBKM Information System & Academic Services using Design Thinking Method (Case Study: Tadulako University),” *IAIC Trans. Sustain. Digit. Innov.*, vol. 6, no. 1, pp. 34–50, Oct. 2024, doi: [10.34306/itsdi.v6i1.676](https://doi.org/10.34306/itsdi.v6i1.676).
- [32] Z. Queen, A. Anjani, and A. G. Prawiyog, “Designing a Digital Business Study Program using Lean Startup Methodology,” *Startupreneur Bus. Digit.*, vol. 3, no. 1, pp. 1–9, Apr. 2024, doi: [10.33050/sabda.v3i1.434](https://doi.org/10.33050/sabda.v3i1.434).
- [33] L. Meria, M. S. Gunawan, S. Solahudin, U. Rahardja, and A. Patel, “Enhancing Digital Business Students Competence through UI/UX Design Training for Digital Product Development,” *ADI Pengabd. Masy. J.*, vol. 6, no. 2, pp. 133–144, May 2026, doi: [10.34306/adimas.v6i2.1467](https://doi.org/10.34306/adimas.v6i2.1467).
- [34] M. A. Setiawan, A. S. Rafika, R. Z. Ikhsan, N. Lutfiani, and R. Evans, “Optimizing the Potential of 3D Printing for Industry from Prototype to Production,” *ADI Pengabd. Masy. J.*, vol. 6, no. 1, pp. 26–37, Nov. 2025, doi: [10.34306/adimas.v6i1.1193](https://doi.org/10.34306/adimas.v6i1.1193).
- [35] E. P. Lestari, A. Valerry, and L. Sulistyowati, “Measuring the Effect of Interdisciplinary Learning Factory Projects on Student Learning Outcomes,” *Int. Trans. Educ. Technol.*, vol. 4, no. 2, pp. 131–148, May 2026, doi: [10.33050/itee.v4i2.1109](https://doi.org/10.33050/itee.v4i2.1109).

- [36] Y. S. Fadillah, I. Yusnita, A. A. Kamal, A. Aprillia, and S. Millah, "Designing an Educational Information System to Enhance Learning Factory Management in Higher Education," *Int. Trans. Educ. Technol.*, vol. 4, no. 1, pp. 66–82, Nov. 2025, doi: [10.33050/itee.v4i1.964](https://doi.org/10.33050/itee.v4i1.964).
- [37] I. K. Mertayasa, U. Rahardja, M. Hardini, and E. P. Harahap, "Interior Design Alphabet Incubator 3.0 Based on Planner 5D," *Technomedia J.*, vol. 8, no. 1, pp. 82–108, May 2023, doi: [10.33050/tmj.v8i1.1982](https://doi.org/10.33050/tmj.v8i1.1982).
- [38] N. P. L. Santoso, R. A. Sunarjo, and I. S. Fadli, "Analyzing the Factors Influencing the Success of Business Incubation Programs: A SmartPLS Approach," *ADI J. Recent Innov.*, vol. 5, no. 1, pp. 60–71, Sep. 2023, doi: [10.34306/ajri.v5i1.985](https://doi.org/10.34306/ajri.v5i1.985).
- [39] A. Padli, N. Khairunnisa, A. Khanza, D. Andayani, and E. Halim, "Strategi Pengembangan Startup Teknologi di Indonesia Melalui IT Business Incubation: Technology Startup Development Strategy in Indonesia Through IT Business Incubation," *J. MENTARI*, vol. 3, no. 1, pp. 73–80, Sep. 2024, doi: [10.33050/mentari.v3i1.621](https://doi.org/10.33050/mentari.v3i1.621).
- [40] R. Ginting Munthe, Q. Aini, N. Lutfiani, I. Van Persie, and A. Ramadhan, "Transforming Scientific Publication Management in the Era of Disruption: SmartPLS Approach in Innovation and Efficiency Analysis," *Aptisi Trans. Manage.*, vol. 8, no. 2, pp. 123–130, May 2024, doi: [10.33050/atm.v8i2.2261](https://doi.org/10.33050/atm.v8i2.2261).
- [41] M. Hardini, M. Mardiana, D. Andayani, E. D. Astuti, and I. Maria, "Evaluation of Digital Branding Strategy in OJS-Based Journals to Increase Dissemination of Scientific Publications," *ADI J. Recent Innov.*, vol. 8, no. 1, pp. 1–12, Sep. 2026, doi: [10.34306/ajri.v8i1.1495](https://doi.org/10.34306/ajri.v8i1.1495).
- [42] S. A. Sibagariang, F. A. Baso, N. Prihantoni, and N. Rangi, "Accelerating Time to Market through Agile in AI Innovation," *J. Comput. Sci. Technol. Appl.*, vol. 3, no. 1, pp. 45–57, Feb. 2026, doi: [10.33050/corisinta.v3i1.158](https://doi.org/10.33050/corisinta.v3i1.158).
- [43] T. S. Agustina, A. T. GENÇ, and A. Prasetyo, "Spiritual Capital Innovative Work Behavior and Subjective Well Being of Startup Founders," *Aptisi Trans. Technopreneurship*, vol. 8, no. 1, pp. 365–376, Mar. 2026, doi: [10.34306/att.v8i1.614](https://doi.org/10.34306/att.v8i1.614).
- [44] P. Rachmadi, R. Situmorang, M. Sukardjo, R. E. Indrajit, and E. Gautama, "Enhancing Entrepreneurial Competencies and Student Engagement through Gamification in Learning Management Systems," *Aptisi Trans. Technopreneurship*, vol. 7, no. 2, Jul. 2025, doi: [10.34306/att.v7i2.589](https://doi.org/10.34306/att.v7i2.589).
- [45] D. Wuisan, J. W. Manurung, C. Wantah, and M. E. Yuliana, "Entrepreneurial Self-Employment and Work Engagement in MSMEs Through Autonomy and Rewards," *Aptisi Trans. Technopreneurship*, vol. 7, no. 1, Mar. 2025, doi: [10.34306/att.v7i1.448](https://doi.org/10.34306/att.v7i1.448).
- [46] Bimantoro, Mulyati, R. Tarmizi, and A. Mutiara Zahrani, "Influence of Incentives and Work Motivation on Employee Productivity at PT Panelindo Graha Nusantara," *Aptisi Trans. Manage.*, vol. 9, no. 2, pp. 176–185, May 2025, doi: [10.33050/atm.v9i2.2467](https://doi.org/10.33050/atm.v9i2.2467).
- [47] I. P. Gustiah, I. Siswanti, K. Kasmir, and L. C. Nawangsari, "The Role of Career Path and Achievement Motivation in Organizational Sustainability from an Entrepreneurship Perspective," *Aptisi Trans. Technopreneurship*, vol. 7, no. 3, Nov. 2025, doi: [10.34306/att.v7i3.687](https://doi.org/10.34306/att.v7i3.687).
- [48] A. Pujiwati, M. Ramaditya, E. Susanty, and S. Prasetyaningtyas, "Workplace Happiness and Retention Strategies for Generation Z in Jakarta Startups," *Aptisi Trans. Technopreneurship*, vol. 8, no. 2, pp. 659–668, Jul. 2026, doi: [10.34306/att.v8i2.741](https://doi.org/10.34306/att.v8i2.741).