Co-funded by the Horizon 2020
Framework Programme of the European Union

HORIZON2020 FRAMEWORK PROGRAMME

ICT – 21 -2014

Advanced digital gaming/gamification technologies

ProsocialLearn

Gamification of Prosocial Learning

for Increased Youth Inclusion and Academic Achievement

D2.4

2nd System Requirements and Architecture

Document Control Page

WP/Task	WP2 / T2.4 System requirements and architecture
Title	D2.4 System Requirements and Architecture (Update)
Due date	30/03/2016
Submission date	22/08/2016
Abstract	This document is D2.4 "2nd Systems Requirements and Architecture" (update) of the Horizon 2020 Project 644204: ProsocialLearn. The document presents the final version of system requirements and architecture for a platform to produce and deliver digital games to teach prosociality to children in educational settings. The document provides a formal description of a system and a detailed plan of the system at component level to guide its implementation. The structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time are described.
Author(s)	Michael Boniface (ITINNOV), Stefano Modafferi (ITINNOV)
Contributor(s)	Simon Crowle (ITINNOV), Lee Middleton (ITINNOV), Jose Miguel Garrido (ATOS), Kosmas Dimitropoulos (CERTH), Kostas Apostolakis (CERTH)
Reviewer(s)	Francesco D'Andria (ATOS)
Dissemination level	☐ internal ☒ public ☐ confidential

Document Control Page

Version	Date	Modified by	Comments
0.1	25/05/2016	Stefano Modafferi (ITINNOV)	First draft
0.2	25/07/2016	Michael Boniface, Jose Miguel Garrido (ITINNOV)	Lesson Management, Overall Architecture updates
0.3	01/08/2016	Michael Boniface, Lee Middleton, Kosmas Dimitropoulos (CERTH), Kostas Apostolakis(ITINNOV)	Player Affect Observation, Performance, and Deployment
0.4	08/08/2016	Michael Boniface, Simon Crowle (ITINNOV)	Learning Analytics
0.5	18/08/2016	Michael Boniface (ITINNOV)	Updates following the review
1.0	22/08/2016	Pilar Pérez (ATOS)	Final and format review

List of Abbreviations

Abbreviation	Description
Active Shape Model	Active Shape Model
AM	Adaptation Manager
API	Application programming interface
ASM	Active Shape Model
AU	Action Units
AuthN	Authorisation
AuthZ	Authorisation
CA	Certificate Authority
CDP	Cloud Design Principles
DCRM	Data Centre resource management
DGM	Deployable Game Model
DN	Distinguished Name
FACS	Facial Action Coding System
GUI	Graphical User Interface
GLM	Game Lifecycle Management
HD	High Definition
IaaS	Infrastructure As A Service
LAN	Local Area Network
LM	Lesson Management
ICT	Information and Communication Technologies
IRI	Internationalised Reference Identifier
LMS	Learning Management System
LOM	Learning Object Metadata
LRS	Learning Record Store
LUT	Look Up Table
MVC	Model View Controller
NPC	Non Player Character

NTP	Network Timing Protocol
OGC	Open Geospatial Consortium
PaaS	Platform as a Service
PAO	Player Affect Observation
PLO	Prosocial learning objective
PSDP	Prosocial Design Principle
PSL	ProsocialLearn
QoE	Quality of Experience
QoS	Quality of Service
RDF	Resource Description Framework
RM	Resource Management
SaaS	Software as a Service
SCORM	Sharable Content Object Reference Model
SPD	Student Performance Dashboard
SSO	Single Sign On
TOGAF	The Open Group Architecture Framework
URI	Uniform Resource Identifier
UUID	Universally unique identifier
VA	Valance Arousal
xAPI	Experience Application Programming Interface

Executive summary

This document presents the final version of system requirements and architecture for a platform to produce and deliver digital games to teach prosociality to children in educational settings. The document provides a formal description of a system and a detailed plan of the system at component level to guide its implementation. The document describes the platform vision providing foresight for the PSL platform and high-level functional overview, the system requirements with an analysis of platform user needs and impacts, the business architecture formalizing stakeholder and domain analysis and use case modelling describing actors, platform interactions and value actors gain from the system, and the information systems architecture formalizing architecture including subsystems, components and interactions.

The high level information systems architecture is summarised in Figure 1. The architecture is composed of loosely coupled services each providing a coherent set of functionality contributing towards management and delivery of prosocial learning experiences. The architecture provides a coherent and consistent description of the capabilities of the platform necessary for end users to understand the requirements and for developers to implement and integrate the final system.

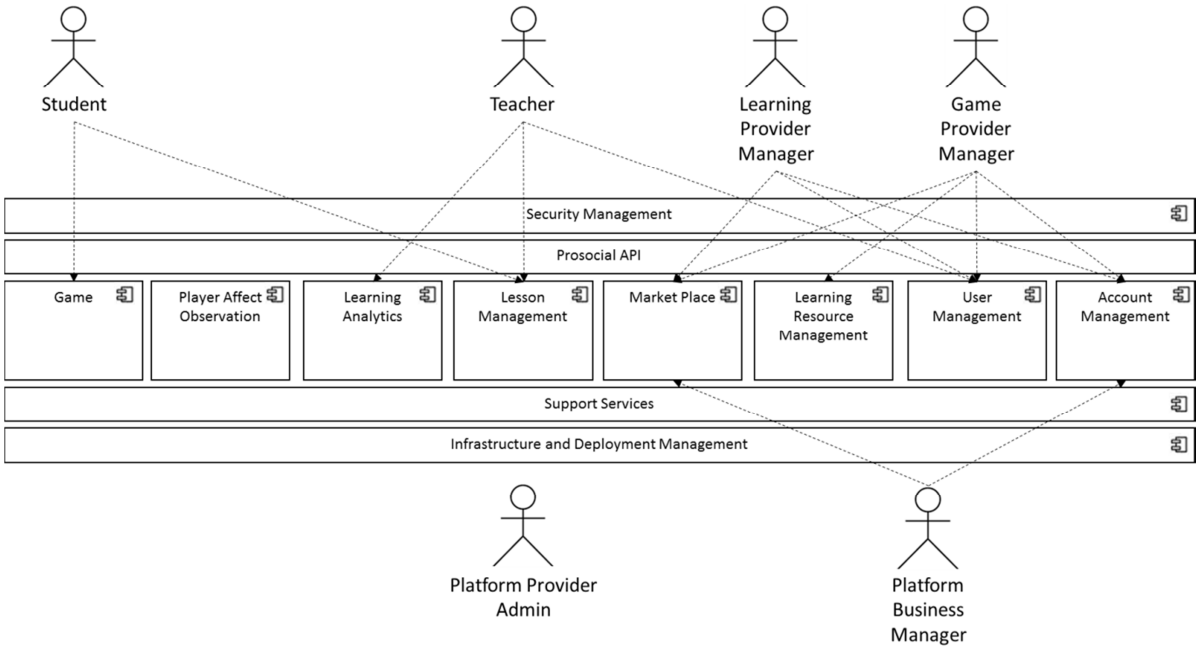

Figure 1: High-level architecture

Index

Table of Figure

1 Introduction

This section provides detailed information about the purpose, scope and structure of the document as well as the intended audience of the document.

1.1 Purpose of the document

This document is D2.4 "2nd Systems Requirements and Architecture" (update) of the Horizon 2020 Project 644204: ProsocialLearn. The document presents the final version of system requirements and architecture for a platform to produce and deliver digital games to teach prosociality to children in educational settings.

The document provides a formal description of a system and a detailed plan of the system at component level to guide its implementation. The structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time are described.

The approach follows parts of the TOGAF® methodology[1] necessary to effectively and efficiently structure communication and implementation in a collaborative innovation project

1.2 Scope and Audience of the document

This document is one of a series of deliverables providing a conceptual view on the gamification of prosocial learning (See Figure 2). The dissemination level of this document is public. This document is the second version of the System Requirements and Architecture superseding the first version D2.3.

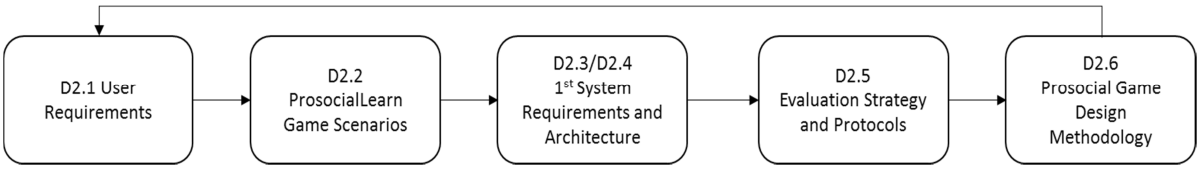

Figure 2: ProsocialLearn document series on the gamification of prosocial learning

In this second version we consider the pedagogical perspective on teaching prosociality and the conceptual framework defined by D2.1 User Requirements V2 along with the findings of D2.6 Prosocial Game Design Methodology which brought together the lessons learnt from the first 12 months of the project and identified the need to increase the emphasis on pedagogical concerns when considering the gamification of prosocial learning.

The documented is scoped according to TOGAF separates analysis into

- Business architecture: describes the requirements of prosocial learn using scenarios and use case modelling
- Information system architecture: defines terminology, and provides a coherent description of the components and conceptual structure of ProsocialLearn's information infrastructure using domain modelling and UML diagrams

The audience targets

- Platform Developers responsible for implementation of Platform Services and Client APIs
- Game Developers wanting to implement prosocial digital games

[1] https://www.opengroup.org/togaf/

- European Commission representatives and reviewers wanting to understand ProsocialLearn's technical approach
- Wider communities of developers and researchers interested in approaches supporting the gamification of learning and the adoption within education systems.

1.3 Structure of the document

The document is structured using a top down approach using the TOGAF framework and analytical methods such as domain analysis and UML modelling. The document includes the following sections:

- Platform vision: foresight for the PSL platform and high-level functional overview
- System requirements: analysis of platform user needs and impacts
- Business architecture: formalized stakeholder and domain analysis and use case modelling describing actors, platform interactions and value actors gain from the system
- Information systems architecture: formalized architecture including subsystems, components and interactions

1.4 Conventions

The keywords "MUST," "MUST NOT," "REQUIRED," "SHALL," "MUST NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

2 Platform Vision

2.1 Gamification of Prosocial Learning

Acquiring skills for social and emotional well-being is important for inclusive societies and academic achievement. Studies have demonstrated the beneficial link between prosocial behaviours and improved results in curriculum topics. The primary purpose of the platform is to enable a Prosocial Learning (PSL) process for creation and delivery of digital games for children (7-10 yrs) within educational systems that support learning of prosocial skills. The approach combines prosocial pedagogies with advanced Information and Communication Technologies and cloud delivery models to create attractive and exciting learning opportunities for children; produce novel digital game-based pedagogies and simplify deployment within educational organisations.

Prosociality is a concept that refers to an individual's propensity towards positive social behaviours. Individuals with prosocial skills are, for example, able to join in conversations, talk nicely, identifying feelings and emotions in themselves and others, identify someone needs help and ask for help. PSL classifies these skills in terms of Friendship, Feelings and Cooperation. By using interactive digital games supported by additional instructive and reflective activities, PSL allows children to learn social skills that can be generalised to real life situations in the classroom, playground and at home.

ProsocialLearn (PSL) project adopts a skills based-approach to learning social skills. We have identified an initial set of 40 skills within three classes: skills for friendship, skills for feelings, and skills for collaboration.

The skills were selected considering their applicability to and benefit from digital game-based learning, for example, the skill can be measured through sensor observation and monitoring tools. The skills are also of different difficulties and can be incrementally learnt to progress students through levels of prosociality. For example, identifying feelings is necessary to be able to showing concern for other's feelings or dealing with angry feelings. Each game can be used to learn one or more skills depending on the nature of the game situations, decisions and mechanics.

2.2 Platform Overview

The adoption of digital games by the formal education sector requires significant innovation in practices of formal schooling, and in the procurement and certification systems for education products. Issues such as the lack of community of practice and time to prepare lessons based on games, the perception that the game takes over from the teacher and incompatibility with teaching practices and general fit to the curriculum are some of the challenges faced by games companies targeting the sector. Our approach to resolving these problems is to provide a platform supporting the multidisciplinary, co-creation of social games in a way that bridges the gap between communities of teachers and game developers, whilst accumulating evidence for the benefits of social games within schools.

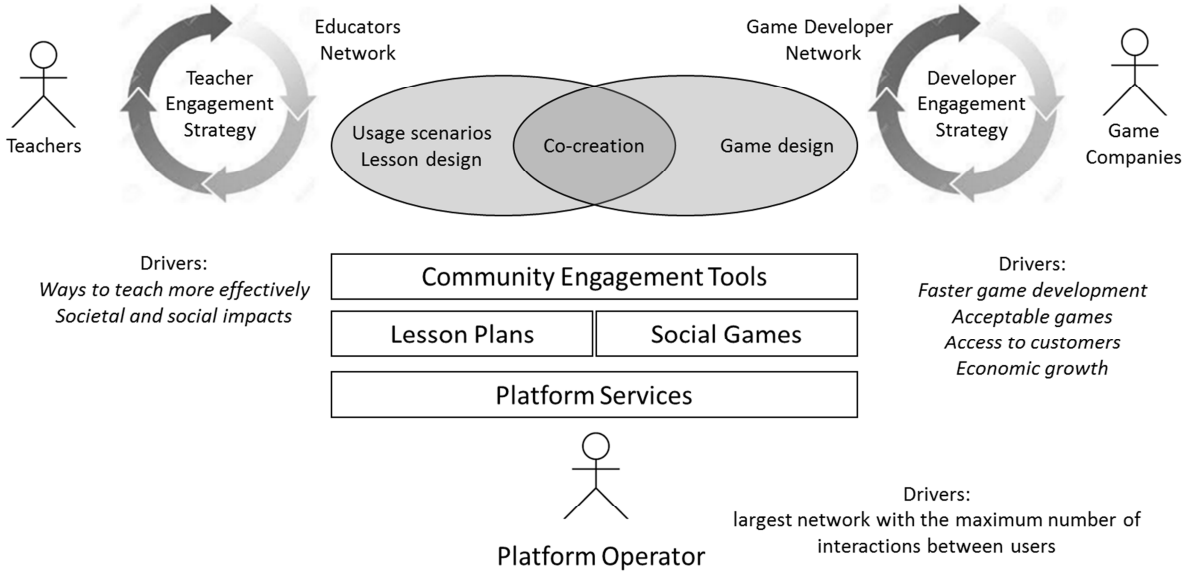

Figure 3: ProsocialLearn platform business concept

Figure 3 shows the ecosystem for the proposed platform. Teachers access games and work with game designers to create and share usage scenarios, lesson designs and success stories within educator networks, promoting adoption and best practice for the wide variety of educational settings expected in schools.

The platform operators business models are based on revenues derived from interactions between users (schools and games companies). The platform itself has little or no value for the initial users and we adopt an appropriate a seeding strategy for engaging users in what is an unknown business proposition initially without critical mass.

The strategy focuses on establishing a set of exemplar games created by teachers and games companies that used to build evidence and are promoted through regional ambassadors. The goal is then to rapidly grow the network on a global scale and remove all barriers to interaction. How the interaction translates to revenue is dependent on the adopted business model. If teachers cannot or will not pay directly (e.g. pay-per-use, subscription, etc.) then other sources of funding are needed e.g. advertising, sponsored games, loss leaders, etc.

2.3 Platform Functional Overview

PSL is implemented through a technology platform offering systematic pedagogical support for prosocial games developed by an ecosystem of teachers and games companies. Capabilities include multi-modal sensors to observe emotional affect, game interaction and decision-making. Information is acquired through standard protocols (e.g. xAPI[2]) and evaluated by learning analytics algorithms to provide real-time feedback on player behaviours that are be used for in-game feedback and adaptation, and by teachers to shape follow-up activities. PSL is deployed at European schools to gather evidence for effectiveness, to promote to policy makers and to increase adoption of game-based learning in schools.

[2] https://github.com/adlnet/xAPI-Spec/blob/master/xAPI.md

The platform considers how to support interactions between different users of the platform (teachers and games companies). Games companies produce value by offering a range of social games to teachers for evaluation and incorporation into lessons.

Games are developed using the platform Prsocial Application Programming Interface (API) that aims accelerates game development by supporting a learning process through generic capabilities for student observation and fusion, game interaction monitoring, social learning analytics, feedback and adaptation and access to the educational game marketplace.

Through these capabilities, games using the platform support a learning analytics pipeline that transforms student monitoring and observations into actionable insights for teachers as part of reflection and feedback activities, or for dynamic intelligent adaptation of the game itself. Three analytics channels are combined to provide knowledge of a player's prosocial behaviour (See Figure 4).

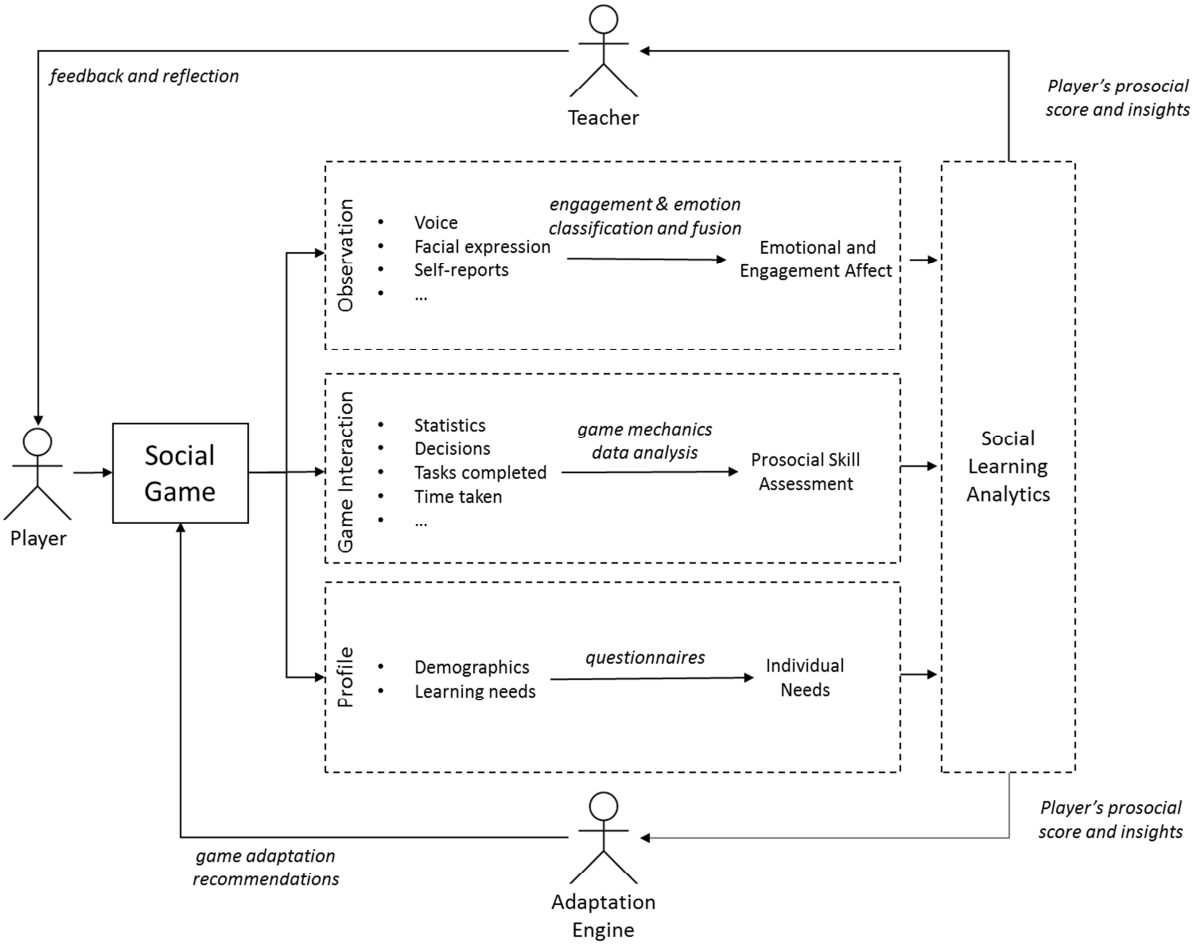

Figure 4: Prosocial learning analytics pipeline

At runtime skill is acquired through game interaction monitoring whilst temporal emotional state is observed through multi-modal sensors analysing voice and facial expression. The data is stored across multiple games as part of a learning record and combined with off-line questionnaires capturing additional information such as demographics and cultural context. The platform aims to

offer capabilities supporting the PSL process through an API offered to games developers and decision support tools for teachers.

3 System Requirements

3.1 User Requirements

User requirements in PSL come from different sources. The market analysis performed in WP1 and reported in D1.1 Market and Competition Analysis[3] sets the scenario where the platform will operate. The user requirement analysis for prosocial learning reported in D2.1 describes a conceptual framework for delivery prosocial games to schools covering aspects of pedagogies, prosocial learning objectives and how they may be related to the gamification of prosocial learning.

The teachers' perspective is captured through the interaction with school representatives and also documented in D2.1. The requirements related to ethical aspects are defined in D7.1[4]. The following sections analyse the impact of the requirements on the architecture identifying a set of technological requirements to be fulfilled by the architecture and its components.

3.1.1 Impact from Market and competition analysis

D1.1 identifies barriers and drivers for the adoption of ProsocialLearn games that impact the system requirements and architecture.

Barriers and drivers
For social games, middleware technologies and platforms are important resource (driver) because they allow developers without in house technologies to produce games that match leisure game quality. Similar middleware technologies on the ProsocialLearn platform could enable developers to produce high-quality games to develop prosocial skills
Scope for a specialist platform to cut production costs. Since collaboration between game designers and professionals in application domains is in its early days, the tools and production process is far from streamlined. Special purpose tools that facilitate game creation for the prosocial skills sector, embedding both game design expertise, management of media assets, evaluation tools, and pedagogical elements can improve the speed and quality of production, and reduce costs
Opportunity to capitalise on downstream restructuring (driver). There are signs that issues related to hosting platforms, distribution, marketing and deployment of digital serious games are being tackled with the aim to structuring and 'pooling', at best in a standardized framework, downstream in the value chain. […] digital serious games is a cross-platform industry.
Opportunity to bring scientific rigour to gaming metrics (driver). While currently products are in the main deployed on personal computers, it will certainly expand onto new generation consoles, and mobile and online platforms. Metrics used to optimize online gaming and maximize revenue can be used instead to evaluate use and behaviour and maximize impact. However this needs to be done in a much more scientific manner with goals of learning, behaviour change etc. that go well beyond customer loyalty or repeat spending, and with considerable care over interpretation."

Table 1: Impact of market on requirements and architecture

[3] ProsocialLearn Consortium. D1.1 Market and Competition Analysis. 2015.

[4] ProsocialLearn Consortium. D7.1 ProsocialLearn ethical oversight procedures. 2015

These impacts are translated in the following set of system requirements. The "m" in the label identifies the requirements is related to "market" needs.

#ID	Description	Priority
m.REQ1	The platform game hosting MUST be offered through a Platform-as-a-Service[5] (PaaS) model where the services are of an ecosystem of different services allows the game developers and the schools to easily find them.	Must
m.REQ2	The platform MUST support Single Sign On (SSO)	Must
m.REQ3	The interaction between Game and the Platform MUST be based on a shared vocabulary explaining prosocial concepts and generic APIs matching game mechanics with prosocial dimensions.	Must
m.REQ4	The platform MUST be scalable to support increasing game demand transparently to game developers.	Must
m.REQ5	The platform MUST provide marketplace where teachers and schools can easily browse, find and procure games offered by games companies	Must
m.REQ6	The platform MUST be flexible to supporting different business models.	Must

Table 2: Technical requirements derived from the market business analysis

3.1.2 Impact from User requirement analysis for prosocial learning

D2.1 defines the user requirements for the gamification of prosocial learning and skill development based on the theoretical understanding of prosociality and its application to the goal of increased youth inclusion and academic achievement.

The second version of the user requirements factored in lessons learnt during the period Apr-15 to Apr-16 to provide a sound and feasible baseline for delivering educational innovation.

The report defines a Prosocial Conceptual Framework as the means to communicate key concepts and theories necessary to for the gamification of prosocial learning in schools.

The framework provides developers (game, game technology, platform) and teaching practitioners clear direction on how to work together to deliver the ProsocialLearn Platform and Prosocial Games.

[5] Platform as a service: https://en.wikipedia.org/wiki/Platform_as_a_service

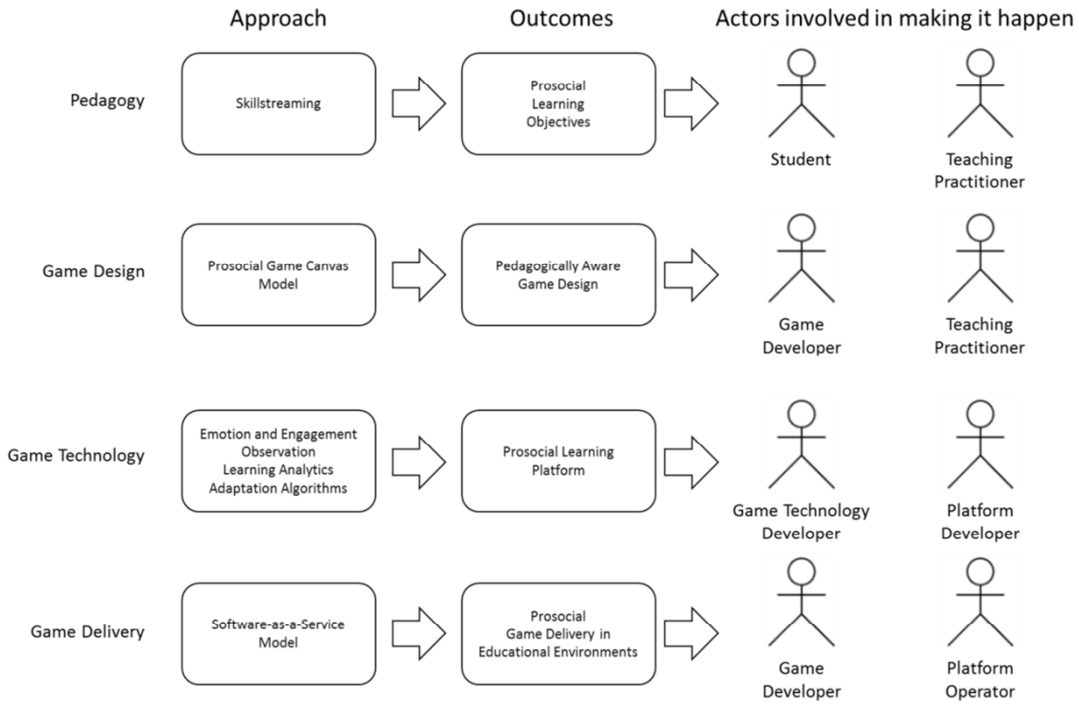

Figure 5: Conceptual Framework for gamification of prosocial learning

The conceptual framework highlights the need for a reliable and accurate process for observing player behaviour and measuring prosocial skills that are analysed to provide real-time adaptation and recommendations to teachers for feedback to students.

The process needs to be supported by a Prosocial API which will be one of the key features of the Platform. The observation and analysis of behaviour and skills implies that the platform will have to handle a significant quantity of data for student assessment by teachers and also the pedagogists aiming to validate the prosocial learn approach.

The implementation will need to handle the richness and accuracy of the models measuring prosociality with specific attention to the intrinsically fuzzy nature of concepts such as emotion and engagement, and how assessment of such quantifies can be linked to prosocial learning objectives. A clear understanding by the game designer of the Prosocial mechanisms is required. Finally it is worth noting that some prosocial learning objectives require multiple prosocial skills which predicate on multiple occurrences of games and therefore a proper behaviour history should be stored and made accessible.

The user requirements are translated in the following set of system requirements. The “p” in the label identifies the requirements is related to “prosocial” needs.

#ID	Description	Priority
p.REQ1	The platform MUST define a clear vocabulary explaining the type of prosocial skills and how they are measured.	Must
p.REQ2	The prosocial observations and game monitoring measurements in the Student Profile SHOULD reported using appropriate standards	Should

	based protocols	
p.REQ3	A game developer MUST describe how a game supports prosocial learning objectives through demonstration of skills within supported game situations	Must
p.REQ4	The platform must store stateful properties related to player affect observations (emotion and engagement) and prosocial skills used and associate these with different game situations.	Must

Table 3: Requirement derived from the User requirement analysis for prosocial learning

3.1.3 Impact from Ethics

A formal procedure for ethical oversight has been defined in D7.1. From a technical point of view the impact of the ethics on the architecture deals with data management and process. ProsocialLearn is a multi-domain environment bring together Learning Providers, Game Providers through an Internet-accessible hosted Platform.

Each of these organisations will operate a distinct security domain and will have responsibilities for security policy within their respective domains. These responsibilities will include:

- Ensure compliance with current laws, regulations and guidelines.
- Comply with requirements for confidentiality, integrity and availability for an organisations employees, students and other users.
- Establish controls for protecting the organisations information and information systems against theft, abuse and other forms of harm and loss.
- Motivate administrators and employees to maintain the responsibility for, ownership of and knowledge about information security, in order to minimize the risk of security incidents.
- Ensure the protection of personal data (privacy).
- Ensure the availability and reliability of the network infrastructure and the services supplied and operated by the organisation
- Comply with methods from international standards for information security, e.g. ISO/IEC 27001.
- Ensure that external service providers comply with the organisation's information security needs and requirements.
- Restrict access to external web sites and services where necessary to mitigate child safety threats

The platform itself must offer various security counter measures to ensure data confidentiality and privacy whilst protecting the services offered by the platform from malicious or accidental threats. Privacy should also been enabled in the platform. In particular it is necessary to:

- support data anonymization, so that the association between the identity records and other information cannot be inferred if data were to get stolen or accidentally disclosed in some way outside the platform;
- provide opt out options that allow specific observation acquisition features (like video or audio) to be disabled on request;
- distinguish between the data accessed by the games hosted on the platform and by the prosocial platform itself that may have access to more accurate and sensitive information upon which it can create a simplified view to be provided to the games; and

- enable user profile browsing only to authorized person like teachers and carers.

The compliance with the data management requirements suggests that it may be easier if processing tasks are moved to the LAN in the Client environment (e.g. within a Learning Provider) rather than processing data streams in the cloud, however, there are trade offers with performance that need to be considered that can limit what's possible. For example, data processing on client devices requires such devices to have significant high end resources computational resources that may not be available in many school environments.

These ethics requirements are translated in the following set of system requirements. The "e" in the label identifies the requirements is related to "ethics" needs.

#ID	Description	Priority [*Must* / *Should* / *Could*]
e.REQ1	All the features provided by the platform related to the acquisition of observations MUST be configurable and it should be possible to disable them on a student-by-student basis.	Must
e.REQ2	A game view on the student profile MUST be defined such that more sensitive information is reserved.	Must
e.REQ3	The platform MUST offer role-based access to the student profiles. Instances of the roles must be related to the student identity (i.e. only a teacher of a given student can access his or her data; that is having the role of teacher does not enable browsing across all available student data.	Must
e.REQ4	Data for student identity and data modelling their interaction between the games and the platform MUST be decoupled, so that they can be related only on the platform, while the association is not possible outside of it.	Must
e.REQ5	The design of the architecture SHOULD enable execution of observation analysis on the client environment where there are restrictions on where sensitive data can be processed.	Should
e.REQ6	Platform users MUST be able to set their privacy settings, by indicating whether their personal information to be used for improving the system and for research. Teachers or Guardians MUST perform this action on behalf of students.	Must
e.REQ7	Platform users SHOULD be able to review the personal information that the system collects.	Should
e.REQ8	Platform users MUST be able to modify and delete personal information, e.g. Teachers or Guardians MUST be able to erase student profiles	Must

Table 4: Requirement from Ethic Management

3.1.4 Impact from the platform users

The PSL platform must support many actors with different roles and different requirements. The requirements derived by the market analysis (cf. Section 3.1.1) partially overlap with this section where we therefore focus on specific requirements not already defined above.

- Game Providers require well-defined and easy to use interfaces and platform performance to be sufficient for not interfering with game experience.
- Teachers want to have clear and possibly easily configurable games; they also require easy interfaces where the prosocial status of students can be assessed and prosocial objectives can be set. Finally they want the possibility of browsing their student profiles.
- Scientists and psychologists need access to aggregated anonymous data for validating the scientific hypothesis underlying ProsocialLearn project.
- Budget holders and decision makers need a clearly designed marketplace where the offer of the platform is highlighted and an easy and flexible way of purchasing games is supported.

The above considerations can be translated into the following requirements. The "u" in the label identifies the requirements is related to "user" needs.

#ID	Description	Priority
~~**u.REQ1**~~	~~Interfaces for game deployment SHOULD be as standard and easy to use as possible~~	Deleted as part of u.REQ2
u.REQ2	The Platform Prosocial API MUST be clear and well documented, stable and the interaction protocol MUST be aligned with accepted standards	Must
u.REQ3	A fast game configurability mechanism SHOULD be made available for teachers	Should
u.REQ4	Teachers and Guardians should be able to browse related students profiles.	Must
u.REQ5	Teachers MUST be able to assess student achievement in terms of Prosocial Learning Objects related to use of prosocial skills.	Must
u.REQ6	It COULD be possible to define objectives as relative increment/decrement of the current assessment	Could
u.REQ7	Scientists and Psychologists MUST to be able to validate the scientific hypothesis underlying the project by having evidence on how the data are collected and having the possibility of browsing the collected data in an anonymous way.	Must
~~**u.REQ8**~~	~~The marketplace should take care of supporting multiple business models so as to suit the biggest possible number of budget holders.~~	Deleted as part of m.REQ6
u.REQ9	A Teacher SHOULD be able to easily consult student profiles through appropriate visual representation of assessment, progress and highlight areas that require teacher attention	Should
u.REQ10	A Teacher MUST be provided with clear explanations on the meaning of student profile items provided upon teacher demand (e.g. by clicking	Must

	on an item).	
u.REQ11	A Teacher MUST be able to set up a lesson for a class in 15 minutes. Ideally games MUST be playable right away or at a scheduled time using default configuration.	Must
u.REQ12	Games MUST be offered as part of a Learning Unit providing a clear summary of the Lesson Plan, other Learning Resources, game play, duration, number of players, Prosocial Learning Objectives (PLOs) and associated Prosocial Skills. A minimum of such information should be required for any game introduced in the marketplace, to facilitate teachers' selection of appropriate games. A default lesson plan MUST be provided by the Game Provider with each Learning Unit. Further Lesson Plans MAY be provided by Teachers themselves and shared with other teachers within a Learning Provider and within the Marketplace	Must
~~**u.REQ13**~~	~~Info for teachers: games should be accompanied by a teacher guide that include suggested lesson plans~~	Deleted as part of u.REQ12
u.REQ14	A teacher guide on prosociality, prosocial concepts and PLOs MUST be available for teacher use and referenced through the games. The platform MUST organize the pedagogical guidelines for teachers and underline the improvements expected over a defined period	Must
u.REQ15	A fast and intuitive setup of student profiles including batch registration MUST be offered that considers the business processes for how students are managed in learning provider organisations	Must
u.REQ16	Student grouping MUST be supported allowing creation of appropriate learning provider structures such as classes, sub classes, groups with specific needs, etc.	Must
u.REQ17	For User and Lesson management functions the Teacher's interface SHOULD be aligned with similar commonly used Learning Management Systems	Should
u.REQ18	The platform MUST deliver fast loading times over standard internet connections: no major waiting either for the platform or the games	Must
u.REQ19	A student profile MUST persists across different groups and teachers within a Learning Provider	Must
u.REQ20	Games SHOULD be incorporated within a typical Lesson schedule suggesting they be completed within 5-25 minutes, or that can reach a meaningful wrap point in 5-25 min	Should
u.REQ21	Games COULD be playable from home and on student's own devices	Could
u.REQ22	Teachers COULD have access to a history of games used and to a personal Library of games they use often	Could
u.REQ23	Teacher MUST be able to rate games and tag games with information about how they used them: e.g. in what learning group, for what	Must

	subject, a small description of their lesson. This information is viewable by other teachers within a Learning Provider or shared within the Community	
u.REQ24	Teachers who do not themselves make purchasing decisions SHOULD have an interface for influencing purchasing decisions, like a wish list	Should
u.REQ25	Teachers SHOULD be provided game recommendations and highlights based on: most commonly used, highest rated, newly added, relevance to other games selected, divergence from other games selected [serendipity principle]	Should
u.REQ26	Teachers and Budget holders MUST be able to search for available games by specific parameters and by word-search	Must
u.REQ27	Students SHOULD be able to rate their enjoyment of games and that information will be in their profiles and available. This information will also be available in the platform in an aggregate form	Should
u.REQ28	Students COULD be able to rate games in terms of their relation to various prosociality concepts and that information will be in their profiles as useful information for the teacher	Could
u.REQ29	Students MUST be able to see an appropriate view of their personal profile.	Must
u.REQ30	The platform and games MUST define clearly all the necessary equipment prerequisites required to deliver learning.	Must
~~**u.REQ31**~~	~~The platform SHOULD organize the pedagogical guidelines for teachers and underline the main aspects of system improvements in a defined period~~	Deleted as part of u.REQ14

Table 5: Requirements from the end-users

4 Business Architecture

4.1 Business Scenarios

4.1.1 Value Proposition Overview

ProsocialLearn is a development and distribution platform used to distribute prosocial digital games from SME game companies to the education sector. ProsocialLearn offers games developers with scientifically proven prosocial game elements that can be used to develop games targeting children for Prosocial Learning.

An application programming interface (API), ProsocialAPI, allows game developers to integrate functions into digital games including visual sensing, identification of prosocial observations from in-game actions, personalised adaptation of game elements, student profiles, game mechanics and expressive virtual characters, and support for data collection with protection of personal data.

ProsocialLearn offers educational institutions a single place to access certified games offering social skills development. Using a pedagogically sound prosocial model that stores student data over time, teachers are able to efficiently monitor progress and assess mid-long term learning outcomes across multiple games and game sessions. Where games are targeting a formal education context they will be described and aligned to curricula helping teaching professionals understand how to incorporate games into teaching programmes.

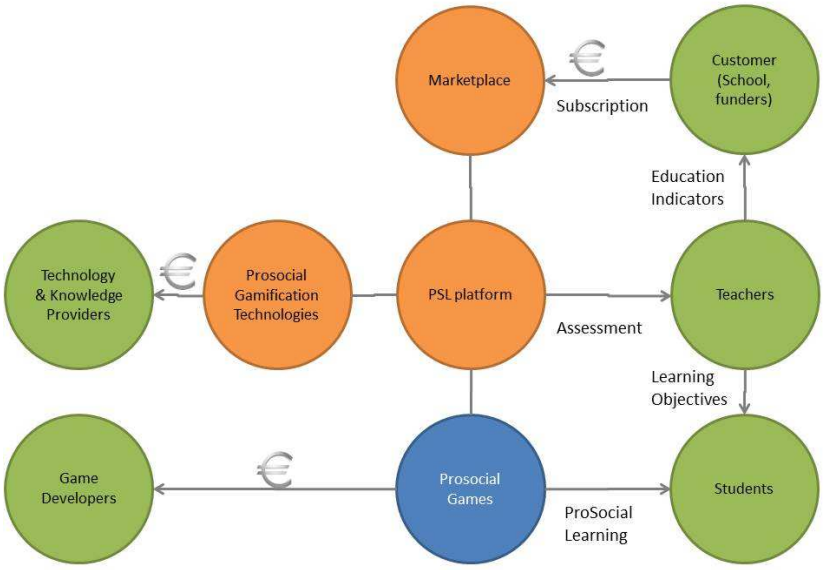

Figure 6: Value Map for ProsocialLearn ecosystem

The value map for ProsocialLearn is shown in Figure 6. Value is expressed in terms of economic and societal transactions. Customers (schools or funding bodies) subscribe to games produced by Game Content Partners that are delivered by the ProsocialLearn Platform.

The Platform builds on Gamification Technologies developed by Technology Providers. A proportion of revenue flows from the Customer to Technology Providers and Game Content Partners. ProsocialLearn will deliver games to Learning Providers using a Software as a Service (SaaS) model.

SaaS provides the means to access European scale markets and lowers the barriers to adoption for Learning Providers by moving capital infrastructure, IT support and upgrade costs to the Platform provider. An SWOT analysis for the ProsocialLearn proposition is given in Table 6.

Strengths	**Weaknesses**
Exciting proposition of using prosocial games to address social inclusion and academic achievement in schools Innovative service delivery model for transferring traditional games to the education sector based on the ProsocialLearn platform Clear vision for architecture and mature technology assets already available as building blocks for the platform Scientifically validated prosocial game elements for certified prosocial games targeting schools Innovative approach for integrating prosocial features into existing game platforms SaaS delivery model to schools Access to large networks of schools throughout Europe for short and longitudinal studies Access to extensive leisure game industry networks World-class multidisciplinary partnership covering all necessary technical, business and sectorial aspects for success An approach that will accelerate research to market Strong SME leadership of business development and game design aspects	Complexity of a pan-European consortium Lack of proof that prosocial gaming in schools can address social inclusion
Opportunities	**Threats**
New European markets for serious games targeting the education sector Single point of access to certified education games New partnerships between leisure games companies and game production and distribution platform provider Reduction in risk to independent gaming companies wanting to develop serious games Lower cost for game companies wanting to produce and distribute games New ways to build serious games based on prosocial features offered as a service New individualised education programmes building on adaptive	Barriers to uptake of digital games within the education sector Insufficient supply from leisure game companies or demand from schools Insufficient scientific validation Mass localisation of content Fragmented customer-based Restricted ICT within school environments Requirements of curricula in

games Acceptance of gamification by teaching professionals and in formal education Scientific validation of effectiveness of the approach	formal education Existing market players with established channels to education sector

Table 6 SWOT Analysis of the ProsocialLearn proposition

4.1.2 Stakeholders

The platform stakeholders are shown Table 7.The stakeholders are individuals and organisations interacting with the platform for a defined benefit.

Stakeholder	Description	Benefits
Student	A person learning prosocial skills, primarily a child of 7-10 yrs old typically learning as part of a group within a classroom setting.	Students are the primary beneficiaries of the PSL platform receiving learning opportunities and experiences.
Learning Provider	An organisation responsible for delivering learning to students typically within an educational setting such as a school.	Access to efficient and effective learning units for teaching prosocial skills in schools.
Game Provider	An organisation responsible for producing learning content including digital games to teach prosocial skills.	Access to game production tools and customers within the education market.
Platform Provider	An organisation responsible for operating and developing the platform infrastructure and growing/supporting the user networks to sufficient scale for sustainable operations.	Monetisation of interaction with learning providers and game providers.

Table 7 Platform stakeholders

4.1.3 Game Provider Scenario

John is the CEO of small Games company called PlayAndLearn. John has decided to sell games to the education sector that teach children prosocial skills. John is a senior person at the organisation with financial responsibility and can commit his company to a business relationship with a PSL Platform Provider.

John registers PlayAndLearn, the PSL Platform Provider receives the registration does some “checks” and approves the PlayAndLearn account.

John can now login to the platform in the role Game Provider Manager. In this role he can add members of his team to the account in the role Game Developer so they can deploy and test games on the PSL Platform. John can also publish deployed games within the PSL marketplace.

John’s team has developed a game called “Helping Your Friends” and want to sell it to the UK education market.

John adds a developer called Alice to his account who created the game. Alice deploys the game on the platform and tests that it works correctly.

Alice describes the game using marketplace metadata so Learning Providers can find the game. Alice let's John know everything is ready.

John publishes the game to the marketplace selecting the Gold subscription pricing model to say that the game can be made available to Learning Providers subscribing to Gold. The PSL Platform Provider does checks on the game to ensure it is appropriate and approves the game in the marketplace

After a month John checks his account and sees that he has earnt £1,667 based on the 80%/20% revenue sharing agreement with the platform provider with 1000 schools paying £500 for 12 month gold subscription for up to 20 games. John withdraws the cash.

4.1.4 Learning Provider

Lilia is the head teacher at a Greek school with budget responsibility. Lilia has a new regulatory requirement to increase social inclusion by teach children social skills as part of the curriculum in 2016/2017. Lilia has heard about the benefits of Prosocial gaming from Teachers at other schools. Lilia registers the school with a PSL Platform Provider who does some checks and approves the School account.

Lilia can now login to the platform in the role of Budget Holder. In this role she can manage her team of Teachers and Administrators

The school is quite large with 1000 students. Lilia adds Dora as a Learning Provider Administrator and a Teacher called Jose. Dora registers all of the students and Learning Groups for the classes in the school. Students can join to several Learning Groups. Dora has created a Learning Group called Dolphin for which Jose is responsible for.

Jose can logins into the platform in the role teacher. Jose wants to create a lesson on Cooperation. He looks at his Lesson Manager and there are no learning resources available to him. He talks to Lilia and she has not purchased any learning units yet.

They look at the options and decided that a Gold subscription is needed to provide access to 20 learning units for up to 1500 children within a Learning Provider organisation. Lilia buys the subscription at £500. Jose goes back to the Lesson Manager and sees a set of 20 learning units.

Jose then creates a new Lesson Plan tailoring the Lesson Plan Template provided with the Learning Unit "Helping Your Friends". John looks at the configuration and accepts the default options of 4 players, maximum game time of 10 mins. John selects the Learning Resource Allocation Plan which randomly allocates students from his Dolphin Learning Group to a game. Jose also makes available some useful assets to set up the Lesson Plan.

These assets include PowerPoint, Word Documents, PDF and links to external resources. Jose then saves his Lesson Plan. He wants 15 minutes to brief the students and will then allow them to play two games of 10 minutes before reflection and debrief. Jose then creates a new Lesson for Monday morning at 10am-11am using his Lesson Plan for a Learning Group he is responsible for. The information is used by the PSL Platform to schedule the resources necessary to support the game.

At 10am on Monday Jose logs into the PSL Platform and selects the Lesson. At 10:15 the students start playing the games. Students login to the platform in the role students and see only the game active at that moment.

After the lesson, Jose checks the Student's Performance using his tablet. For each student he can see if the students have demonstrated sufficient cooperativeness during the game and the emotional affect acquired from their voice.

He noticed that one of the students Dimos increased his cooperativeness in the second game, and noted to praise Dimos during the next lesson. The lesson was a success so Jose decides to share is Lesson Plan with all teachers in the School. If they agree it's a good plan he will publish it more widely to the PSL community in the marketplace.

4.2 Actors

The platform actors within each stakeholder organisation are shown in Figure 7.

Each actor represents one or more roles that an individual Account User can fulfil. All platform users beyond those within the Platform Provider itself exist within the context of an Account. The Account represents a trust relationship between the Platform Provider and different types of stakeholder organisations interacting with the platform.

Each Account is owned by a responsible party within a stakeholder organisation including Research Manager, Learning Provider Manager and Game Provider Manager. A full description of each actor role is described in Table 8.

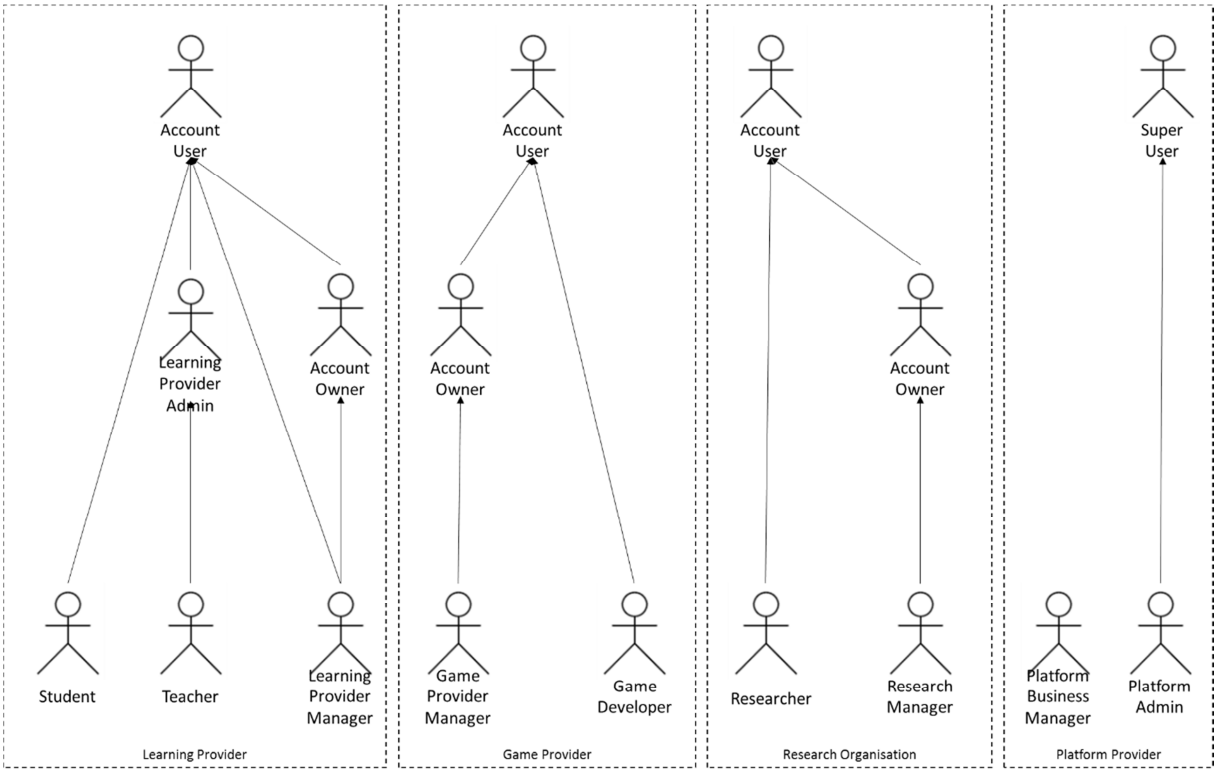

Figure 7: PSL actors

Actor	Description
Account User	A person registered with a PSL Platform within an Account Owner's organization

Account Owner	A person with authority to commit a stakeholder to a relationship with the Platform Provider under the terms and conditions of the account type
Super User	A person at a Platform provider who can perform all administration actions on the platform
Platform Manager	A person responsible for managing the business of the PSL Platform
Platform Provider Admin	A person responsible for the management of the PSL Platform
Student	A person learning prosocial skills, primarily a child learning as part of a group within a classroom setting
Teacher	A person teaching prosociality to individual or groups of students through lessons that incorporate learning resources based on digital games
Learning Provider Admin	A person responsible for managing groups of students and employees at a Learning Provider
Learning Provider Manager	An Account Owner at a Learning Provider organisation
Game Provider Manager	An Account Owner at a Game Provider organisation
Game Developer	A person responsible for deploying and testing learning resources at a PSL Platform Provider
Research Manager	An Account Owner at a Research Organisation
Researcher	A person responsible for analysing student performance data and establishing scientific evidence for the benefits of PSL

Table 8: PSL actor descriptions

4.3 Domain Model

The high level platform domain model is shown in Figure 8 showing the primary conceptual elements and their relationships. A Learning Provider Manager gets access to Learning Units via the Marketplace by registering for a Learning Provider Account and signing a Learning Provider Access Agreement giving sufficient access permissions.

The Learning Provider Access Agreement could be a subscription or pay per use agreement depending on the platforms business model. The Learning Units themselves are owned and published by a Game Provider Manager who makes them available on the Marketplace on the basis of a Game Provider Publishers Agreement.

Each of the Agreements outlines the financial basis for accessing the Marketplace. For Learning Provider's a Pricing Model is defined by the Platform Business Manager to calculate bills from Learning Unit Usage. For a Game Provider a Revenue Model is defined by the Platform Business Manager to calculate earnings from Learning Unit Usage.

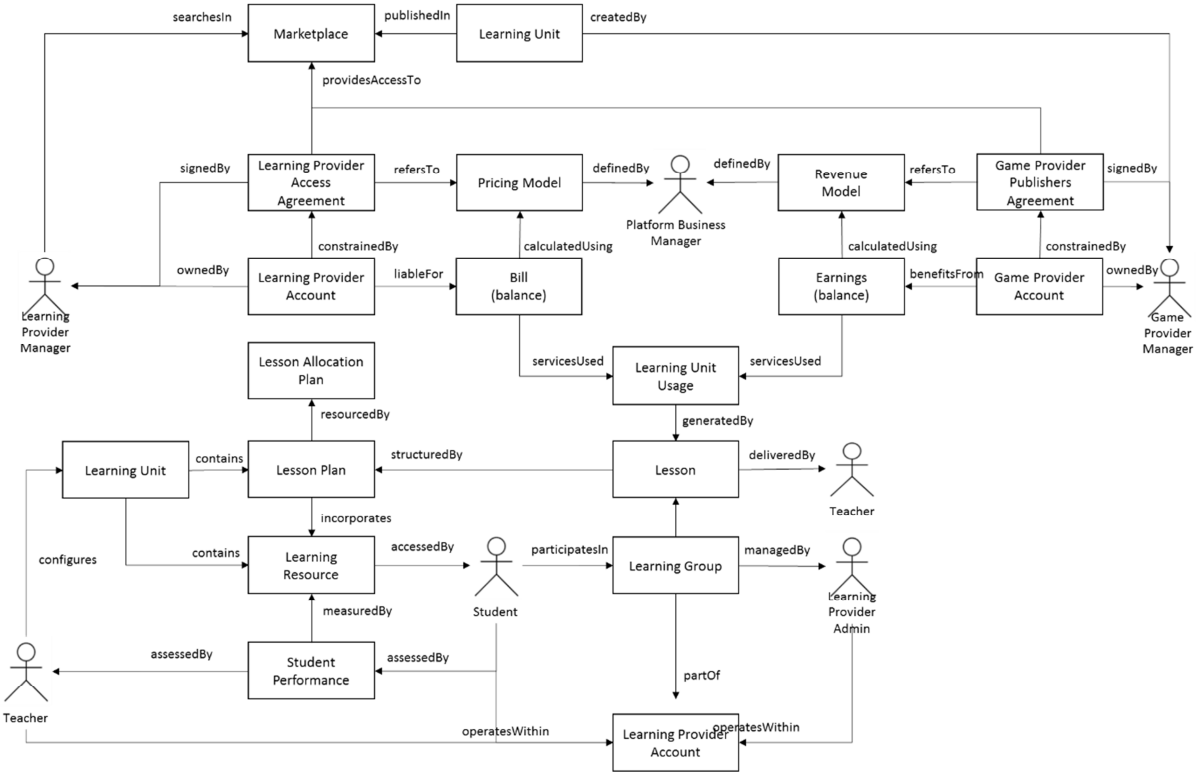

Figure 8: High level domain model

Learning Units are used and configured by Teachers to create concrete Lesson Plans for Learning Groups whose membership consists of sets of Students.

The Lesson Plan includes a Resource Allocation Plan for allocating students within a Learning Group to Games themselves. When a Lesson Plan is executed a Lesson is created delivering a period of learning to the Learning Group. During the Lesson students access Learning Resources including a digital game to teach prosocial skills. Student Performance is measured and assessed by the Teacher to provide reinforcement feedback and to determine the level of attainment for each student.

A description of each concept within the high level domain model is given in Table 9.

Resource	Description
Learning Unit	A unit of assessment defining prosocial objectives and goals students are expected to achieve describing key ideas, concepts and topics.
Learning Resource	A game and other digital resources that form part of a Learning Unit necessary to teach prosocial skills
Learning Resource Usage	A measure of the use of a Learning Resource used to calculate Bills and allow a Platform Provider to understand market dynamics

Marketplace	An online e-Commerce where Learning Units are published, searched and sold
Learning Provider Account	The root of trust between a Learning Provider and the Platform Provider
Learning Provider Access Agreement	An agreement between a Learning Provider and Platform Provider defining the terms, conditions, rights and obligations of each party in relation to services offered, primarily access to Learning Units within the Marketplace
Bill	Money owed by a Learning Provider for services provide by a Platform Provider
Pricing Model	An algorithm applied to Learning Unit Usage to calculate a bill
Game Provider Account	The root of trust between Game Provider and the Platform Provider
Game Provider Publishers Agreement	An agreement between a Game Provider and Platform Provider defining the terms, conditions, rights and obligations of each party in relation to services offered, primarily access to the Marketplace for publication of Learning Units
Earnings	Money earnt by a Game Provider for usage of Learning Units
Revenue Model	An algorithm applied to Learning Unit Usage to calculate earnings
Learning Group	A group of students at a Learning Provider (e.g. a class)
Lesson Plan	A generalised procedure and configuration template for delivering a Lesson to a Learning Group based on a Learning Resource
Learning Resource Allocation Plan	A strategy for allocating Learning Resources to students within a Learning Group.
Lesson	A period when individual Students or Learning Groups learn prosocial skills by accessing Learning Resources in accordance with a Lesson Plan
Student performance	A measure of achievement of prosocial skills

Table 9: High level domain model concepts

The game domain model is shown in Figure 9 showing the conceptual elements for the gamification of prosocial learning. Games themselves form part of Learning Units along with a Lesson Plan Template and other supporting Learning Resources such as PowerPoint and printable media. The Lesson Plan template provides a detailed description of the course of instruction that can be tailored by teachers for specific Learning Groups.

This includes default game configuration of offline and real-time adaptation mechanisms and a Lesson Allocation Plan for distributing members of Learning Groups to Game instances. Basic allocation plans include random and fully customisable allocation, although further plans based on different criteria such as performance levels can be envisaged.

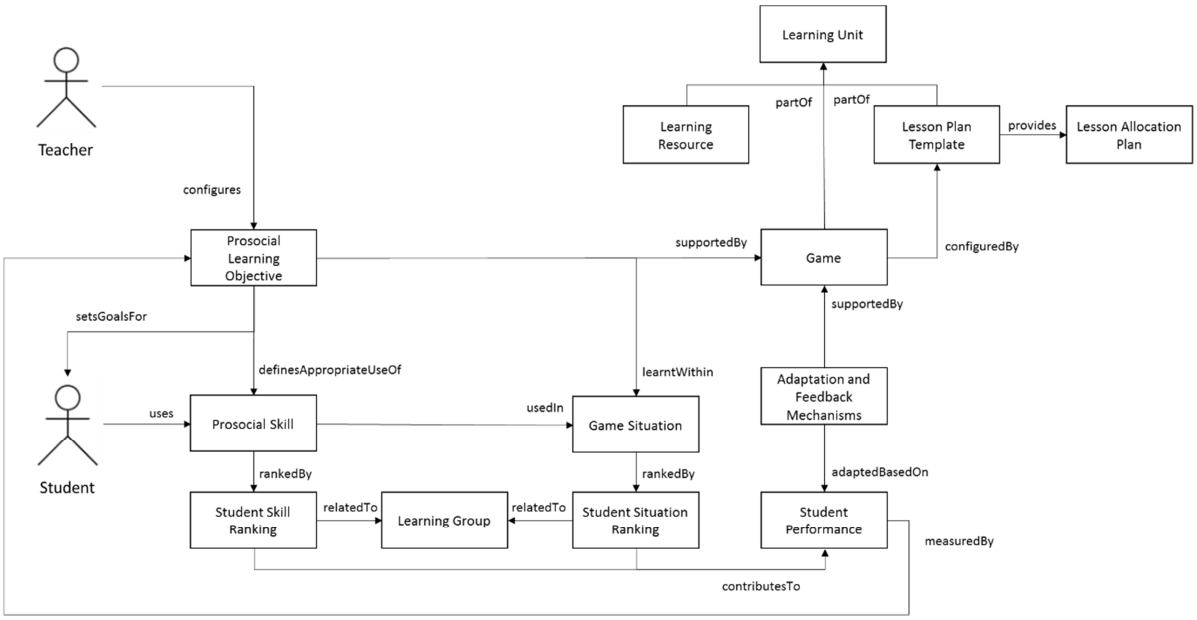

Figure 9: Game domain model

Each Game supports one or more Prosocial Learning Objectives (PLOs) by allowing a Student to use Prosocial Skills within different Game Situations.

Each PLO defines the appropriate usage of a skill (e.g. sufficient, insufficient, etc.) for a specific student in a given situation. Students are ranked in relation to their Learning Groups in terms of use of the skill and also the use of a skill in a specific Game Situation. The ranking is used to calculate student performance as a way of measuring progress towards a PLO and as input to Adaptation and Feedback mechanisms providing offline and real-time game adaptation as a way of maximising learning potential. The ranking is normalised by Learning Group based on historical data for the group and is essentially a Leaderboard.

A description of each concept within the game domain model is given in Table 10

Resource	Description
Learning Resource	A digital media object within a Learning Unit needed to support the delivery of a Lesson (e.g. PowerPoint slides, PDFs, etc)
Lesson Plan Template	A detailed description of the course of instruction used to guide student learning with points of adaptability so lessons can be customised for specific Students and Learning Groups
Lesson Allocation Plan	An algorithm for distributing members of a Learning Group to Game instances.
Game	A digital game for learning Prosocial Skills
Prosocial Learning Objective	An assessment mechanism that allows a Teacher to measure student achievement within the parameters of a particular academic standard (i.e. prosociality)

Prosocial Skill	The ability and appropriate use of prosocial behaviours in different situations
Game Situation	A situation where students and practice and demonstrate the use of Prosocial Skills
Adaptation and Feedback Mechanism	A point of variability in a Game Situation that can be configured offline and/or at real-time
Student Ranking	A score that measures student performance in relation to other students within a Learning Group (i.e. a leaderboard) for a specific Prosocial Skill
Student Situation Ranking	A score that measures student performance in relation to other students (i.e. a leaderboard) within a Learning Group for a specific Prosocial Skill used in a specific Game Situation
Student Performance	An aggregate score that measures student performance in relation to a Prosocial Learning Objective

Table 10: Game concepts

4.4 Use Case Model

The use case model defines the interactions between roles (actors) and the platform to achieve specific goals.

4.4.1 Use case groups

Four use case groups are defined grouping functionality of the platform into logical sets as shown in Figure 10, with descriptions of each group in Table 11

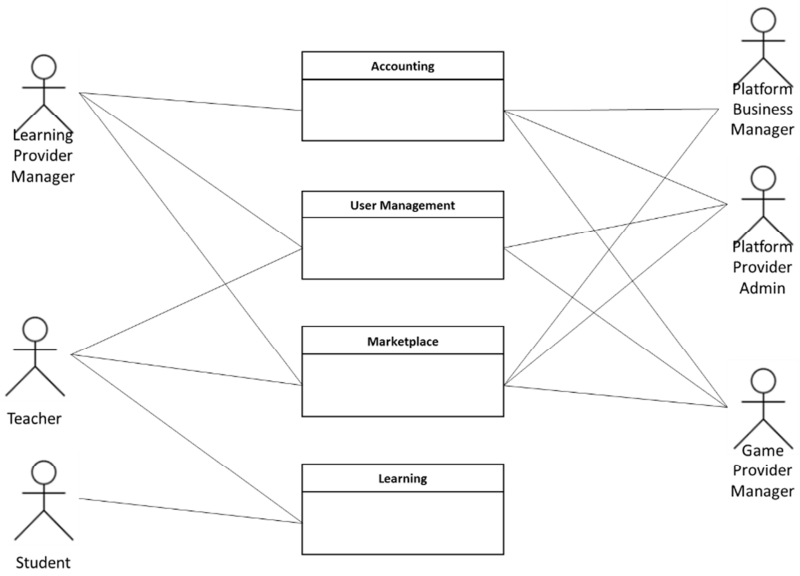

Figure 10: Use case groups

Use Case Group	Descriptions

Accounting	Interactions related to managing the business relationship (root of trust and billing) between the PSL Platform Provider and Account Holders (roles: Learning Providers and Game Providers) that use the PSL Platform
User Management	Interactions related to the management of user roles within Learning Provider (roles: Teacher, Student Admin, Student) and Game Provider (roles: Game Developer) organisations
Marketplace	Interactions related to deploying, publishing, searching and buying Learning Resources
Learning	Interactions related to the delivery of Lessons to Students by Teachers, and the assessment of Student Performance.

Table 11: Description of use case groups

4.4.2 Accounting Use Cases

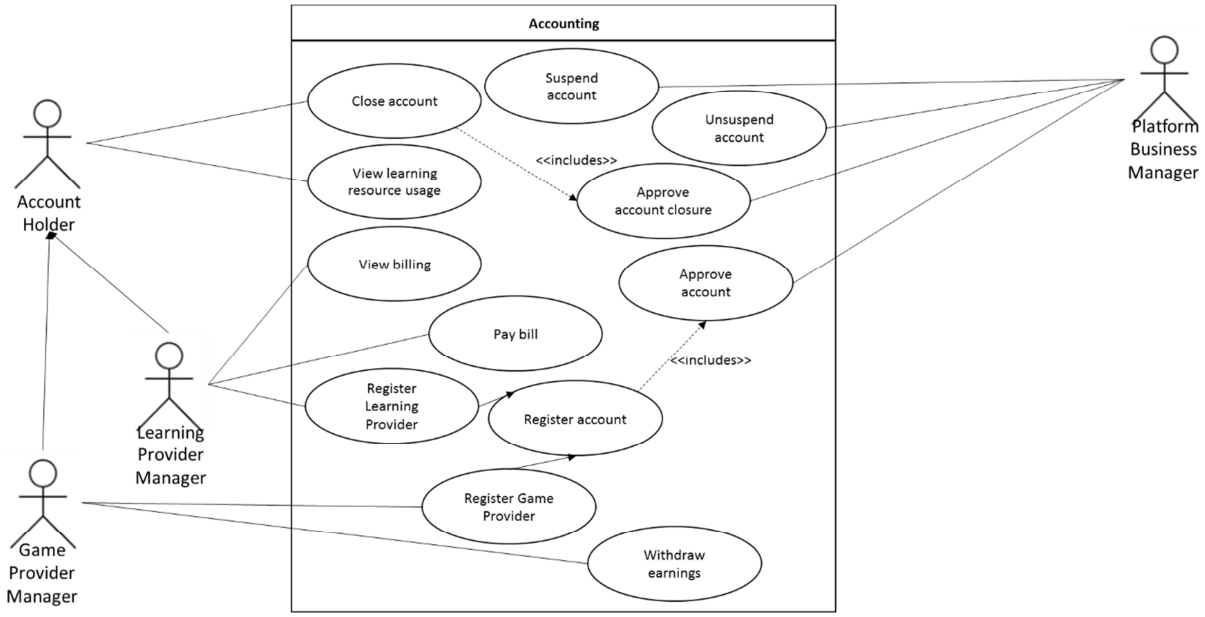

Figure 11: Accounting use cases

Use Case Group	Outcomes
Register learning provider	A Learning Provider account is created. A Learning Record Store instance is created for the account A user with the role Learning Provider is created who is owner of the account
Register game provider	A Game Provider account is created A Learning Resource storage location is created for the account A user with the role Game Provider is created who is owner of the account
Suspend account	Access to all resources associated with an account is denied

Close account	Access to all resources associated with an account is denied All resources associated with an account are archived or deleted depending on policy
View learning resource usage	The usage of learning resources is displayed
View billing	The money owed on an account is displayed to an Account holder
Withdraw earnings	A transaction transferring the money earnt by a Game Provider to an allocated bank account is completed

Table 12: Summary of accounting use case outcomes

4.4.3 User Management Use Cases

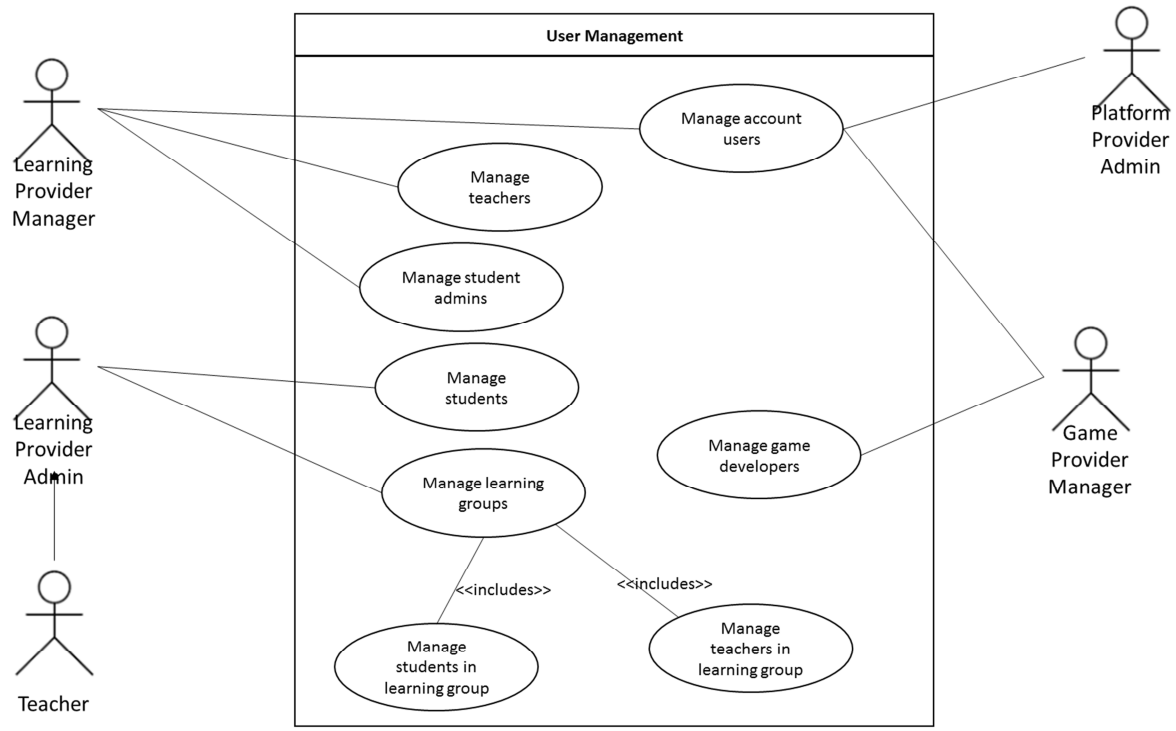

Table 13: User management use cases

Use Case Group	Outcomes
Manage account users	CRUD users associated with an Account
Manage teachers	CRUD account users with the Teacher role
Manage student admins	CRUD account users with the Student Admin role
Manage students	CRUD account users with the Student role
Manage learning	CRUD Learning Group resources

groups	
Manage teachers in learning groups	CRUD account users with the Teacher role able to manage Students within a Learning Group
Manage students in learning groups	CRUD account users with the Student role participating within a Learning Group
Manage game developers	GRUD account users with the game developer role

Table 14: User management use case outcomes

4.4.4 Marketplace Use Cases

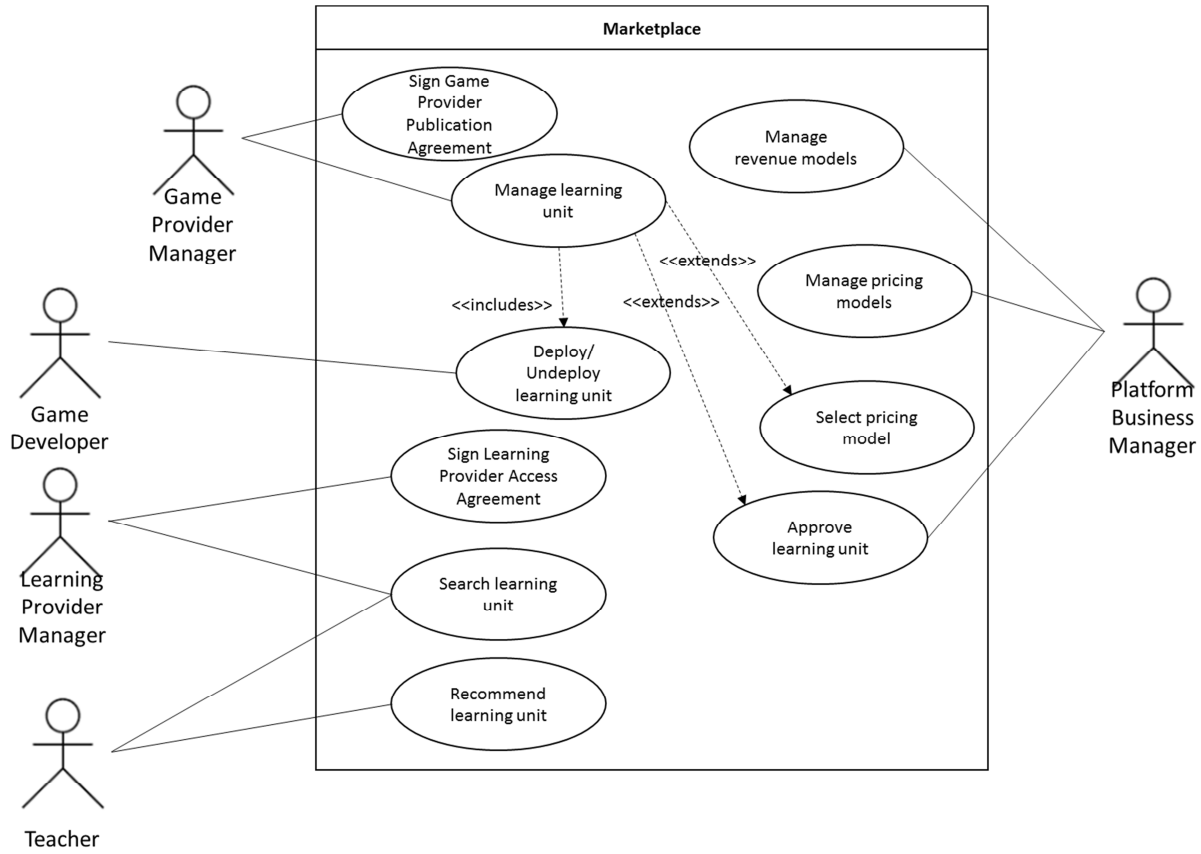

Figure 12: Marketplace use cases

Use Case Group	Outcomes
Manage pricing models	CRUD ways by which Game Providers can price Learning Units when publishing to the Marketplace
Manage revenue models	CRUD ways by which Game Providers receive revenues from Learning Units when publishing to the Marketplace
Manage learning units	CRUD Learning Units available for search and buying by Learning Providers as part of Learning Provider Access Agreements

Sign learning provider access agreement	Learning Units that form part of the terms and conditions are available for use by Teachers who have access to the Learning Provider Account, and the Learning Provider is liable for any monies owed in accordance with the applicable pricing model
Sing game provider publication agreement	A Game Provider account has permission to manage Learning Units, publish them in the marketplace subject to approval and receive earnings according to the applicable Revenue model.
Search learning units	A list of Learning Units filtered according to search criteria
Recommend learning unit	A Learning Unit recommended by a Teacher for purchase by a Learning Provider Manager

Table 15: Marketplace use case outcomes

4.5 Learning Use Cases

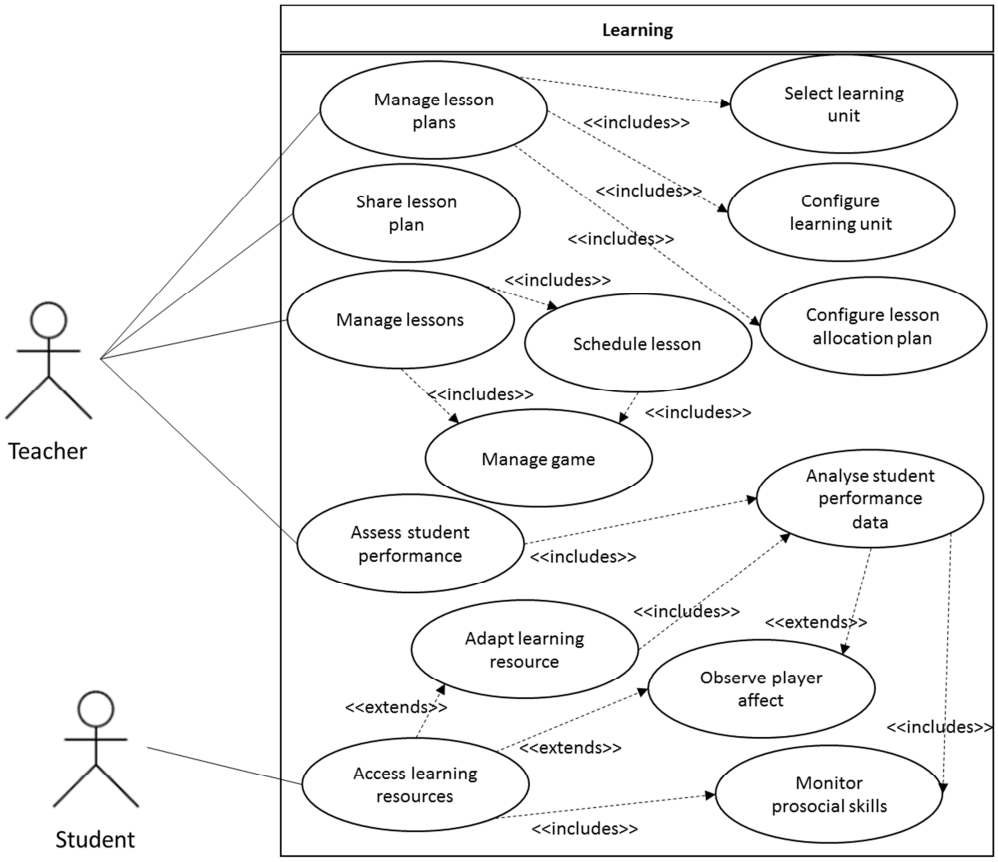

Figure 13: Learning use cases

Use Case Group	Outcomes
Manage lesson plans	A generalised procedure and template for delivering a Lesson

Share lesson plan	A lesson plan that is published within the market place and associated with relevant learning units.
Manage lessons	CRUD lessons based on games scheduled to start either immediately or at some future time
Access learning resources	A list of games accessible and playable by Students during Lessons
Assess student performance	Student performance measured in accordance with prosocial learning objectives and assessed by Teachers to provide feedback and made available for adaptation of learning

Table 16: Learning use case outcomes

5 Information Systems Architecture

5.1 Overview

The high level information systems architecture is shown in Figure 14. The architecture is composed of loosely coupled services each providing a coherent set of functionality contributing towards management and delivery of prosocial learning experiences. Each subsystem offers APIs that contribute to the full Prosocial API with respective data management and user interface components. A description of each subsystem is shown in Table 17.

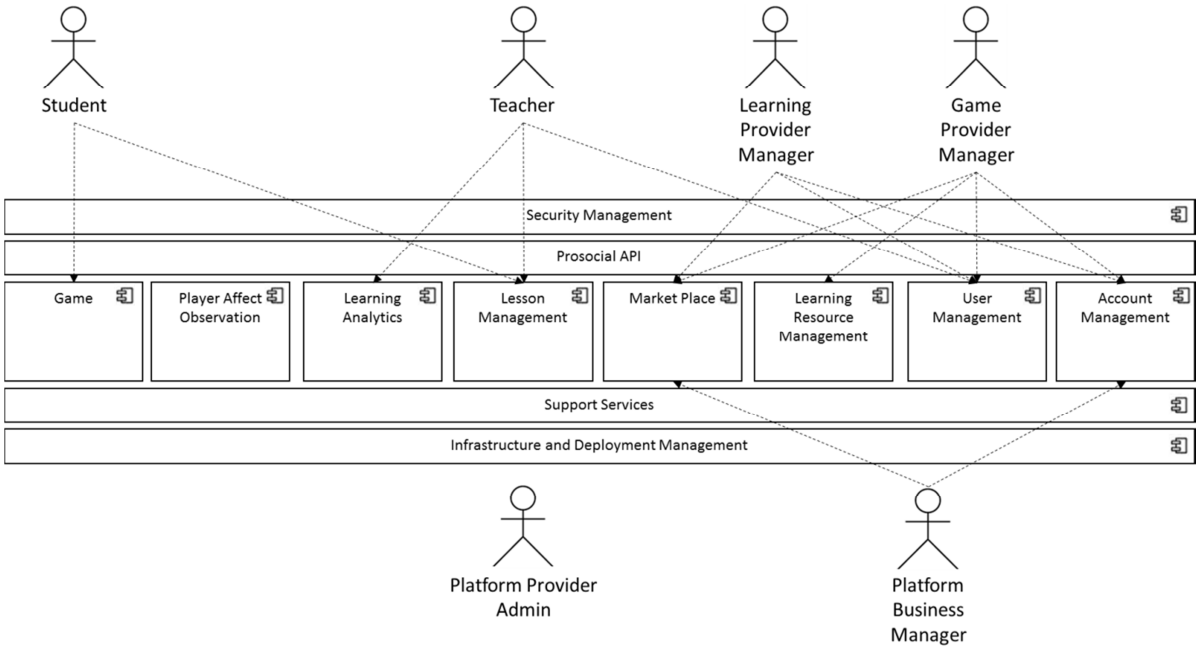

Figure 14: High-level architecture

Subsystem	Functional description
Account Management	Provides function to establish business relationships between platform users and the platform provider
User Management	Provides function to manage platform user roles and associated groups
Learning Resource Management	Provides function to manage the lifecycle of learning resource artefacts including deployment, testing, and certification.
Marketplace	Provides function to publish, search and buy learning units
Lesson management	Provides function to manage lessons based on learning units
Learning Analytics	Provides function to assess students learning performance during lessons and retrospectively
Player Affect Observation	Provides function to observe and measure prosocial indicators such as emotion and engagement
Game	Provides function to play digital games that are designed to teach prosocial

	skills
Security Management	Provides function to authenticate and authorise users and their agents (e.g. services, client applications)
Support Services	Provides function for core distributed system capabilities such as load balancing, service registry, time synchronisation and content management
Infrastructure and Deployment Management	Provides function to manage infrastructure required to host platform and game services
Prosocial API	The Prosocial API provides a document developer guide for programming the PSL platform building on the APIs of the platform subsystems

Table 17: Platform subsystem descriptions

5.2 General Principles

5.2.1 Naming

Naming is concerned with the rules for choosing identifiers to denote applications, software, data, people and things. This includes services and the resources they provide access to.

Naming must consider the scope and relative uniqueness of identifiers. Naming syntax and conventions are especially important in distributed systems where things are interacting with, shared with or being observed by multiple system components often developed and operated independently.

The architecture uses **Convention** rather than **Resolution** for naming through agreement on the names to describe entities prior to execution based on international standards where possible. There are situations where prior agreement cannot be done because either the entities are not known at the start or existing naming conventions are already established.

The architectural principles for naming are:

- Identifiers are assigned when an entity is born by a component responsible for creating them (e.g. the LessonManager component creates LessonManagementId)
- Identifiers should be unique enough so that they do not clash in a context of use (e.g. globally, learning provider, lesson, etc).
- Identifiers should be structured according to URIs where possible.
- Identifiers based on URIs can be dereferenceable but this is not mandatory.
- Identifies based on internationally accepted standards (e.g. xAPI[6], SCORM 4th Edition[7], IEEE 1484.12.1-2002 Learning Object Metadata- LOM[8], Open Geospatial Consortium (OGC) Observations and Measurements v2.0/ISO/DIS 19156[9])

There architecture considers the following cases where consistent naming is essential:

[6] https://github.com/adlnet/xAPI-Spec

[7] https://www.adlnet.gov/adl-research/scorm/scorm-2004-4th-edition/

[8] https://standards.ieee.org/findstds/standard/1484.12.1-2002.html

[9] http://www.opengeospatial.org/standards/om

- Individual users, roles, groups, and organisations
- Sensor observation (e.g. emotion and engagement) and in-game monitoring data (e.g. prosocial skill use and scores) are observations about students collected during a game sessions.
- Learning Unit metadata describing prosocial skills, game types, game situation types, adaptation mechanisms.
- Service instances and associated resources such as Lesson instances, Game instances, etc.

Throughout the remainder of the document each subcomponent includes a naming convention table describing the resource, component responsible for creating the resource, the naming convention and the scope of the name. For example, the naming convention for a learning group is given below:

Resource	CreatedBy	Convention	Scope
Learning Group	User Management	IFI e.g. <platform-url>\lg\UUID according to the Group/IFI:account ID type from the xAPI specification e.g. *www.prosociallearn.eu\lg\cd5f371a-4fe3-48ed-bcc7-c9f2c1cad42e*	Learning Provider

Table 18: Example naming convention for a learning group

5.2.2 Prosocial Vocabularies

The platform components are described using well-known and approved vocabularies to ensure effective communication of capability between platform users.

Such vocabularies include those needed to describe Prosocial Skills, generalised pedagogical elements such as adaptation mechanisms and feedback that are implemented in the games and platform itself, and units of measure of sensor measurements related to prosociality.

The metadata describing the Learning Units and other parts of the platform must refer to the approved vocabularies so descriptions are accessible by Platform Users in a consistent form. The terms are defined and managed by a Prosocial Learning Authority either as a group within the Platform Provider or as an external accredited body responsible for certification of prosocial learning units.

The naming conventions for prosocial learn vocabularies are shown in Table 19.

Resource	CreatedBy	Convention	Scope
Prosocial Skill Vocabulary	Prosocial Learning Authority	A de-referenceable URI <platform-url>\skill\<name> according to the IEEE Standard for Learning Technology—Data Model for Reusable Competency Definitions[10] using the "Long identifier type". Prosocial Skills are unique among all contexts in which the skill may be used. The Catalog element is the namespace for all prosocial skills. The Entry element is the unique identifier of the skill within the catalog.	Global

[10] http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4445693

		e.g. www.prosociallearn.eu\skill\helpingothers	
Sensor Observation and Measurement Vocabulary	Prosocial Learning Authority	A de-referenceable URI <platform-url>\uom\<name>. No relevant standard identified. Measurement types are unique among all contexts in which learning is being conducted. *e.g.* *www.prosociallearn.eu\uom\valance,* *www.prosociallearn.eu\uom\arousal,* *www.prosociallearn.eu\uom\engagement*	Global
Adaptation Mechanism Vocabulary	Prosocial Learning Authority	A de-referenceable URI <platform-url>\adaptation\<name>. No relevant standard identified to date. Adaptation types unique among all contexts in which learning is being conducted but should be shared between game implementations *e.g. www.prosociallearn.eu\adaptation\difficulty*	Global
Feedback Mechanism Vocabulary	Prosocial Learning Authority	A de-referenceable URI <platform-url>\feedback\<name>. No relevant standard identified has been identified to date. Feedback types are unique among all contexts in which learning is being conducted. *e.g. www.prosociallearn.eu\feedback\reenforcement*	Global

Table 19: Naming for platform vocabularies

Vocabularies are published on a web server hosted at the platform domain currently www.prosociallearn.eu:

5.2.3 Time Synchronisation

The platform provides time synchronisation for observation and measurement of student performance along other key events within the lifecycle of learning. Synchronisation is supported:

- to correlate measurements between students playing in the same game instance on different client devices. For example, when a teacher wants to explore the different student interaction and outcomes through visualisation of data.
- to correlate measurements between student interaction, game lifecycle management, adaptation and feedback events. For example, when a teacher wants to explore the impact of online adaptations on students within a specific game instance.

Time synchronisation for data correlation and visualisation is within 100mS allowing relationships between game events and student affect to be analysed.

Synchronisation between multi-modal observation streams of an individual student (e.g. emotion and engagement) for data fusion is not required as these measurements are performed on a single client device that itself is synchronised.

A platform Timing Service is provided that provides a Platform Time to platform services and game client applications. The Timing Service is hosted on a machine that calibrated with a reliable timing

source using Network Timing Protocol (NTP). The Timing Service operates in master/slave mode allow clients to synchronise their time to the server using either HTTP or WebSockets. These protocols allow synchronisation of javascript game clients accessing the platform through a firewall.

Services and game clients are synchronised either periodically or at well-defined events within the lifecycle (e.g. when a game is created). Clock drift is not a significant issue as game instances are unlikely to run beyond the length of a typical lesson of 1 hr in duration.

The timing service algorithm is detailed as part of the timesync documentation[11].

5.3 Platform Subsystems

Figure 15 shows a detailed component model breaking each subsystem into components. Each component is described in more detail in the subsequent sections.

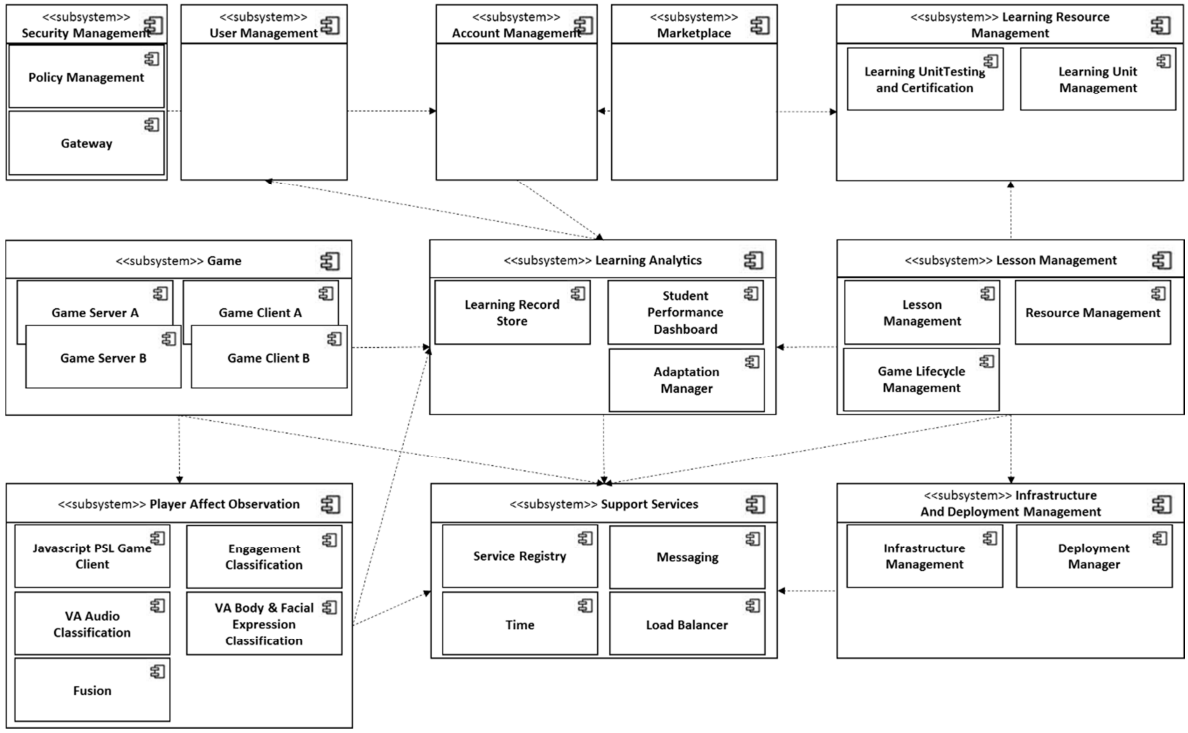

Figure 15 Architecture Component Model

5.3.1 Account Management Subsystem

The account management subsystem provides function to manage the business relationship between platform user organisations and the Platform Provider. This includes primarily the Learning Provider and Game Provider but also Research Organisations.

The relationship is manifested through an account resource that is owned by an Account Owner from the user organisation. The naming conventions for the subsystem are given in Table 20.

Resource	CreatedBy	Convention	Scope

[11] https://www.npmjs.com/package/timesync

Game Provider Account	Account Management	IRI <platform-url>\name according to the Group/IFI:account ID type from the xAPI specification *e.g. www.prosociallearn.eu\gp\funandlearnco*	Global
Learning Provider Account	Account Management	IRI <platform-url>\name according to the xAPI Group Account specification *e.g. www.prosociallearn.eu\lp\londonschool*	Global
Bill (debt transaction)	Account Management	A unique transaction ID asserted by account management *e.g. 4534536*	Platform
Earnings (Credit Transaction)	Account Management	A unique transaction ID asserted by the accounting system *e.g. 657657*	Platform

Table 20: Naming conventions for the Account Management subsystem

When an organisation wants to access the platform the Account Owner requests an account with the Platform Provider specifying the organisation type either Learning Provider, Game Provider or Research Organisation.

The existence of an account allows the Account Owner to manage their relationship with the Platform Provider in accordance with their organisation role. In general terms, an account allows the owner to manage user access within their organisation, managing end user agreements with the platform provider for service access and manage usage accounting and billing in accordance with those agreements and applicable revenue and pricing models.

The subsystem allows a Platform Business Manager to manage relationships with platform users including approval and revocation of accounts, reporting of usage and billing.

5.3.2 User Management Subsystem

The User Management subsystem manages users, their roles and associated groups

5.3.2.1 User Registration and Identifiers

User identify is a process managed by the Platform within the User Management component. Users obtain an identity by registering with the platform within the context of well-defined business processes. When a user organisation wants to access the platform an organisation account is created to establish a business relationship between platform user organisations and the platform provider.

A Budget Holder with decision making authority is responsible for requesting an account at which point the platform provider will perform due diligence to authenticate that the Budget Holder who they say they are. This may include out of bound checks including phone calls, email authentication or providing valid financial information.

On approval of the account the Budget Holder is provided with an identify that allows them to manage users within their organisation. A Platform user exists within the context of a single organisation account.

Budget Holders, Teachers, Learning Provider Admins users at Learning Provider organisations and Game Developers are all identified through an email address based on FOAF mbox identifiers in accordance with the xAPI specification.

Students identify is managed differently to 1) separate personally identifying information (e.g. first name and last name) from learning records and 2) to simplify login for children. Each student is identified by

- a UUID prefixed with the platform IRI responsible to creating the identify http://prosociallearn.eu/pids that is globally unique
- a memorable string representing visual character shape:colour:animal that is scoped within the context of an organisation account

Students do not manage their own identify but are registered within the platform by a Learning Provider Admin.

Learning Provider Admins manage a pool of virtual characters for their organisation that are assigned to students when they join the Learning Provider. On assignment the Student gives their virtual a secret name that's used as their password. The virtual character is kept by the Student for the period when they are attending the Learning Provider, typically multiple years. When a Student leaves the Learning Provider the Memorable string can be reused for other students. A combination of 10 colours, 10 shapes and 20 animals gives an active student population of 2000 which is sufficient for larger Learning Providers.

5.3.2.2 User Password

Budget Holders, Teachers, Learning Provider Admins and Game Developers possess a string password to authenticate using the HTTP Basic authentication scheme as defined in [RFC2617] to authenticate with the authorization server.

Students possess a secret name for their visual character that is used as the password.

5.3.2.3 Roles and Groups

The roles of platform users are represented through a set of groups.

Subject	Description
Student Group	The collection of students within a Learning Provider Account
Teacher Group	The collection of teachers within a Learning Provider Account
Learning Provider Admin Group	The collection of teachers within a Learning Provider Account
Learning Provider Manager	An individual authenticated user from a Learning Provider organisation responsible for the Learning Provider Account
Game Provider Manager	An individual authenticated user from a Game Provider organisation responsible for the Game Provider Account
Game Developer Group	The collection of game developers within a Game Provider Account
Learning Group	A collection of students learning prosocial skills
Platform Client	A collection of trusted platform component clients such as Adaptation Manager, Classification

Platform Operator Group	A collection of platform administrators within a Platform Operator

Table 21: Platform roles represented through subject groups

A Learning Group is important contextual information that forms part of a student's learning record.

As such Learning Group's are identified using IFI <platform-url>\lg\UUID according to the Group/IFI:account ID type from the xAPI specification. For example, www.prosociallearn.eu\lg\cd5f371a-4fe3-48ed-bcc7-c9f2c1cad42e

5.3.3 Learning Unit Management Subsystem

The Learning Unit Management subsystem provides functionality to manage Learning Unit Resources provided by Game Providers.

Game Providers provide Learning Unit packages that include game software and other learning resources, and describe the package using standardised metadata. The Learning Unit Package is created in accordance with SCORM content package as a Package Interchange File (PIF) and it contains all files needed to deliver the content package on the platform.

The file is zip format. Although the platform does not use a SCORM runtime engine alignment with SCORM for packaging brings interoperability benefits in terms of publication and search. The Learning Unit package structure is shown in Table 22.

Path	**Description**
root\imsmanifest.xml	Describes the content package including the resources in the package and the metadata describing the learning according to IEEE LOM.
root\game	Game software
root\game\server	One or more Docker images required by the game server
root\game\server\install.yml	An ANSIBLE script in YML format for composing/installing the game server using the document images
root\game\client	Any client application software distributions
root\resources\	One or more digital resources supporting the learning unit

Table 22: Learning Unit Package Structure

Besides that, Game Providers can themselves directly deploy games in the virtualized environment managed by the PSL platform (with an easy to use drag and drop functionality) or just provide a link to the Game installed on their own infrastructure.

Game software is deployed as a Docker images to allow the Platform Provider to deploy and host game instances in accordance with resource management policies necessary to fulfil their Learning Provider Access Agreement commitments.

A SCORM manifest file is used to describe the Learning Unit. The manifest must include metadata according to IEEE Learning Object Metadata model with at least the following elements shown in Table 23.

LOM Element	**Comments**

General	Used to describe general information about the learning unit including title, languages supported, geographies, etc.
Lifecycle	Used to describe versioning and attribution information. Must be consistent with the Learning Unit Identifier
Technical	Used to describe the technical requirements for deploying the Learning Unit including any prerequisites for Learning Providers in terms of IT infrastructures
Educational	Used to describe the pedagogical aspects of the Lesson Plan Template.
Classification	Used to classify the Learning Unit in accordance with the Prosocial Skill Vocabulary.

Table 23: IEEE LOM Usage for Game Packaging

The naming conventions for the learning unit management system are shown in Table 24.

Resource	Naming Convention	Scope
Learning Unit	URI <platform-url>\learningunit\UUID\<version> according to IEEE LOM identifier. Version MUST be in the form 1.0.0 and be consistent with the lifecycle element of the LOM metadata • Major version, UX changes, file format • Minor features, major bug fixes, etc. • Minor bugs, spelling mistakes, etc *e.g.www.prosociallearn.eu\lu\5ec1e016-afee-4d18-8d64-572db6b2be38\1.1.2*	Global
Learning Resource	Resource identifier string within the SCORM manifest file, unique within the learning unit. *e.g. <resource identifier="resource_1" type="webcontent" adlcp:scormType="sco" href="shared/launchpage.html">*	Learning Unit
Lesson Plan Template	Resource identifier string within the SCORM manifest file, unique within the learning unit *e.g. <resource identifier="resource_2" type="webcontent" adlcp:scormType="sco" href="shared/launchpage.html">*	Learning Unit
Game Type	URI of published game <platform-url>\game\<name>\version SCORM resource identifier within the manifest file, unique within the learning unit but also globally. Version MUST be in the form 1.0.0 • Major version, UX changes, file format • Minor features, major bug fixes, etc. • Minor bugs, spelling mistakes, etc *e.g. www.prosociallearn.eu\game\FindWithFriends\1.1.2*	Global
Game Situation Type	URI <platform-url>\game\< name>\<situation-name> Could be encoded as a SCORM item but as we are not using the runtime engine then it is in appropriate to encode beyond the learning unit level using	Game

	SCORM. *e.g. www.prosociallearn.eu\game\FindWithFriends\situtation1*	
Game Situation Instance	A UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) according to Activity IDs in the xAPI specification *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	Platform

Table 24: Naming Conventions for Learning Unit Management

After a Game Provider has uploaded a Learning Unit Package it is available for deployment, testing and certification.

5.3.4 Learning Unit Testing and Certification Subsystem

The Learning Unit Testing and Certification subsystem provides functionality to ensure that learning content is robust, reliable and suitable for teaching prosocial skills to students. The subsystem provides a testing infrastructure to allow game providers to verify and validate functionality, along with mechanisms to transfer Learning Units to a production infrastructure for beta trials and finally publication within the marketplace.

The Platform Provider provides a test infrastructure along with representative data sets for Learning Providers that can be used to test the games. In addition the Platform Provider offers access to a community of Teachers who are interested to participate in closed and open beta programmes.

Integration and Alpha tests are deployed on a test infrastructure with anonymised data whilst beta tests and certification are deployed on the production infrastructure to allow for scalability of the system and to ensure that student's data collected as part of the beta test is handled according to applicability security policies for data protection and confidentiality. Descriptions of the testing phases are shown in Table 25.

Testing Phase	**Description**	**Infrastructure**
Integration	The purpose of integration testing is to create a working system that allows the interaction between the game and the platform.	Test
Alpha	The purpose of alpha testing is to improve the quality of the learning unit to ensure beta readiness. This happens towards the end of the development process when the learning unit is in a near fully-usable state. A Game Provider can expect there to be many iterations and it is expected alpha phases to be 3-5x the length of beta. The focus is quality/engineering (bugs, bugs, bugs) and is performed by test engineers, employees, and sometimes "friends and family". The Learning Unit is expected to have bugs, crash, missing docs and features. The outcome will be that most known critical issues are fixed, some features may change or be added as a result of early feedback, and the Game Provider will have a good idea how the game will perform.	Test

Closed or Open Beta	The purpose of beta testing is to improve the quality of the Learning Unit by integrating input from Learning Providers, and ensure release readiness. The Game Provider will to beta testing just prior to launch in the marketplace and will take usually only a few weeks (sometimes up to a couple of months) with few major iterations. The activity will involve product marketing, support, docs, quality and engineering to test in the "real world" with "real customers" and the feedback can cover every element of the Learning Unit. Much of the feedback collected is considered for and/or implemented in future versions of the Learning Unit. Only important/critical changes are made as a consequence. The outcome of beta tests is to understand the limits of a Learning Unit by allowing Learning Providers to explore in their environments. The Game Provider will have a good idea of what the Learning Provider thinks about the Learning Unit and what s/he is likely to experience when they purchase it.	Production
Certification	The purpose of certification is to ensure that the Learning Unit is suitable for teaching prosocial skills to students. No specific certification authority has been decided but candidates could include the Platform Provider themselves or external authorities such as: https://www.kidsafeseal.com/aboutourprogram.html https://www.pshe-association.org.uk/quality-mark-for-resources	Production

Table 25: Learning unit testing phases

After certification of a Learning Unit a game provider can publish the Learning Unit to the marketplace.

5.3.5 Marketplace Subsystem

The marketplace subsystem provides function to publish, search and buy learning units incorporating prosocial games. The marketplace is managed by the Platform Business Manager who is responsible for defining marketplace economics through definition of pricing models, revenue models, game provider publication agreements and learning provider access agreements.

Game Providers publish learning units to the marketplace according to a Game Providers Publication Agreement.

The agreement outlines the outlining terms, conditions, obligations and liabilities for Game Providers such as revenue models and any specific market constraints that apply (e.g. geographies). When an agreement is signed it is associated with a Game Provider Account providing permissions for users associated with the account to perform actions on the platform such as developers testing Learning Units, publishing for certification and making marketplace offers.

A published game is described according to the Learning Unit metadata specification (see Section 5.3.3) allowing Learning Providers to discover new learning content, to understand potential learning outcomes and how the content can be integrated into the classroom and school IT infrastructure.

Learning Providers search the marketplace for learning content using Learning Unit metadata search terms. A Learning Provider can access Learning Units by agreeing to a Learning Provider Access Agreement. The agreement outlines terms, conditions, obligations and liabilities for Learning Providers such as pricing models and usage constraints (e.g. how many students, etc).

When an agreement is signed it is associated with a Learning Provider Account providing permissions for users within the account to perform actions on the platform such as create lessons using a Learning Unit covered by the agreement.

The naming conventions for market place resources are shown in Table 26.

Resource	CreatedBy	Convention	Scope
Learning Provider Access Agreement	Marketplace	A numerical key representing a transaction ID asserted by the marketplace *e.g. 3546373*	Platform
Pricing Model	Marketplace	UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) *e.g. c204edc1-5d80-4deb-bcb3-faae98a677ab*	Platform
Revenue Model	Marketplace	UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	Platform
Game Provider Publishers Agreement	Marketplace	A numerical key representing a transaction ID asserted by the marketplace *e.g. 3546373*	Platform

Table 26: Naming convention for marketplace resources

5.3.6 Lesson Management Subsystem

The Lesson Management subsystem is concerned with managing the lifecycle of lessons delivered to Learning Groups within a Learning Provider organisation. The subsystem is responsible for ensuring that all platform resources necessary to deliver the learning experience are provisioned and available for the period of learning

Resource	CreatedBy	Naming Convention	Scope
Lesson	Lesson Management	A UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) according to Activity IDs in the xAPI specification *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	Learning Provider
Prosocial Learning	Lesson Management	A UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) according to Activity IDs in the	Student

Objective		xAPI specification *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	
Game Instance	Game	A UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) according to Activity IDs in the xAPI specification *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	Platform

Table 27: Naming for the Lesson Management Subsystem

5.3.6.1 Creating a Lesson Plan

A Teacher describes the Lesson requirements through a Teachers dashboard by selecting a Learning Unit and tailoring a Lesson Plan Template (provided by the Game Provider) for a Learning Group. This process includes tailoring group and individual level prosocial learning objectives, selecting an allocation plan for distributing students in the group to game instances, and setting any game specific configuration if needed. The lesson is given a start time and duration which is initially taken from the Lesson Plan metadata.

The platform expects this process to be done prior to the Lesson and to take no longer than 15 minutes.

5.3.6.2 Lifecycle Management

The platform uses the requirements defined in a Lesson Plan to provision resources for a Lesson. This includes tasks to configure required platform services, provisioning and management of game instances and automatically allocating students to those instances. The platform provides a fully automated procedure to minimise teacher administration tasks therefore increasing their productivity.

The Lesson Management (LM) component incorporates business processes to orchestrate Lessons on behalf of Teachers. The LM informs a Scheduler component of Lesson Plans. The Resource Management (RM) component is responsible for managing resources necessary to deliver the aggregated set of Lessons on the platform. The RM monitors service instances and performance targets through interaction with resource management systems Cloud Management and Load Balancer. The RM uses the Deployment Manager to schedule deploy and undeploy service instances and game Dockers as required, and to set performance targets for load balancers (e.g. classifiers).

The LM uses the Game Lifecycle Management (GLM) component to provision and configure game instances for Lessons. Game instance management is achieved using a Game Management API implemented by Game Servers allowing the platform to automate the allocation of Learning Groups members to Game instances.

The LM provides the GLM with a Lesson plan and the GLM, using a Game Management API, initiates the creation of game instances on a game server and allocation Students to those game instances. The GLM implements the allocation algorithms for associating students to game instances based on requirements in the Lesson Plan (e.g. Random or full customised) and considers the constraints of the game itself (e.g. the number of players for game).

Each Game Instance has platform configuration information that details information on service endpoints that is published to the Service Registry. The Game Server looks up the configuration in the registry using the Game Instance ID.

The platform configuration is in JSON format and includes the elements described in Table 28.

Elements	Description
Policy Management	Endpoint for the service providing AuthN and AuthZ
LRS xAPI	Endpoint for the service providing storage of xAPI assertions about student performance.
Timing Service	Endpoint for service providing synchronisation of the game server with platform time
VA Classification Services (or load balancer)	Endpoint for services providing player affect observation of emotion
Engagement Classification Services	Endpoint for services providing player affect observation of engagement
Adaptation Manager	Endpoint for service providing real-time adaptation of game situations
Messaging	Endpoint for publishing student activity state and game state for monitoring by the platform

Table 28: Game instance configuration information

The Game Management API is a restful protocol specification including the following operations:

- GameInstanceID : Create()
- Start(GameInstanceID)
- Reconfigure(GameInstanceID)
- Pause(GameInstanceID)
- Stop(GameInstanceID)
- Destroy(GameInstanceID)

The Game Management API supports different control scenarios such as time-based games and synchronised starting between multiple players. The game server itself is responsible for enforcing the control scenarios and reporting management events to the platform via the Messaging component.

GLM handles student exceptions (e.g. player disconnects, player is sick for a scheduled lesson, toilet break, fire alarm, etc) by managing and monitoring the activity of a student and allowing a teacher (or automatically) to take action such as adding new student, pausing the game, etc. Monitoring player activity state is reported by the Game Server to the GLM via the Messaging component.

5.3.7 Player Affect Observation Subsystem

The Player Affect Observation (PAO) subsystem automatically acquires, classifies and fuses player emotion and emotion in relation to game play events. A series of multi-modal observation channels are established from input sensors connected to player devices including microphones, cameras and keyboard, in addition to more advanced gaming sensors such as Kinect and Leap Motion. Using sensing and classification techniques emotion and engagement from voice, facial expression and body language is be acquired and then fusion processes applied to provide temporal emotional and engagement state.

Individual classifiers are used for each of emotion and engagement. The classifiers are supervised and pre-trained on an input set of labelled features. The individual classifier outputs are then fused to make a final decision based upon the input features. Fusion happens at two specific levels: feature level fusion and decision level fusion. Feature level fusion takes place within the classifiers. The platform supports a number of different feature detectors where the detectors generate features in a high dimensional space. When the feature vectors are combined from different modalities there is a risk of high misclassification rates. The platform uses dimensionality reduction and feature selection techniques to reduce this risk.

Dimensionality reduction embed a high dimensional space into a lower one. Normally, implicitly assumes that the data lies on a manifold in the high dimensional space. Non-linear dimensionality reduction techniques such as Laplacian Eigenmaps or locally linear embedding are preferred. Feature selection use search and an evaluation metric to evaluate the feature set. These techniques build an alternate representation of the data. Approaches such as clustering, support vector machines, and naïve Bayes are preferred. The platform uses a combination of these where appropriate to the underlying features. Fusion at this level will allow the exploitation of dependencies within features. These algorithms are understandably complex and effort will be made to reduce the computational overhead.

Decision level fusion is performed based upon the outputs from the individual classifiers. The benefit of this second fusion step is primarily to provide robustness to missing observations. The combination allows the possibility of estimates of confidence based on the pooled output of the individual classifiers. The approaches used include aggregate and statistical methods. The favoured approaches here are either aggregate or statistical methods as the method of fusion is comparatively simple and has minimal computational overhead.

Player affect observations are modelled in accordance with the OGC Observation and Measurement model according to the JSON implementation profile[12].

Valance-arousal dimensional space is used for all modes of measurement unit for classification of emotional state[13] (. This allows a discretisation process to split the space and cover all the different emotions required for all the different input modalities and allows classifiers to use the same labels and apply basins of attraction around emotions concepts we can cluster them together.

OBSERVATION	FEATURES OF INTEREST	MEASUREMENT	MEASUREMENT UNIT
Voice	Carious sound features (energy in frequency band, pitch, duration)	Audio stream capture, feature extraction classification	Valance-Arousal
Facial Expression	Low-level distance and angle measures associated with eyes, eyebrows and mouth	Video stream capture, feature extraction and classification	Valance-Arousal
Self-Reported Emotions	Feedback provided by the user on perception of valence and arousal	Selection of valance and arousal from in-game emoticons	Valance-Arousal

[12] http://www.opengeospatial.org/standards/om

[13] Russell, J.A. (1980). A Circumplex Model of Affect. Journal of Personality and Social Psychology 39 (6): 1161–78

Table 29: Emotion Observation Model

The sensor observation channels are incorporated into a data fusion pipeline using decision-level techniques to bring together multi-modal sources and analyse emotional affect. Decision-level fusion is used to determine events along a timeline at which learned decisions are extracted from all unimodal features and integrated into higher semantic concepts (e.g. valance arousal).

The platform enables multichannel acquisition of observations. Each channel has its data processing pipeline related to the nature of the observation but are integrated into the platform using a consistent architectural model to allow for manageability.

Each classifier implements a stream-based processing function that processes sensor data input and asserts player affect observation output for downstream processing and analytics (e.g. Adaptation and Learning Record Store).

The distribution of the data processing function across different components is not dictated by the platform as it is dependent on resourcing and data protection constraints, and associated tradeoffs.

For example, classifying video on a client device reduces network bandwidth and avoids the need to transfer personally identifying information to the platform but moving processing to clients increases the technical requirements for such devices.

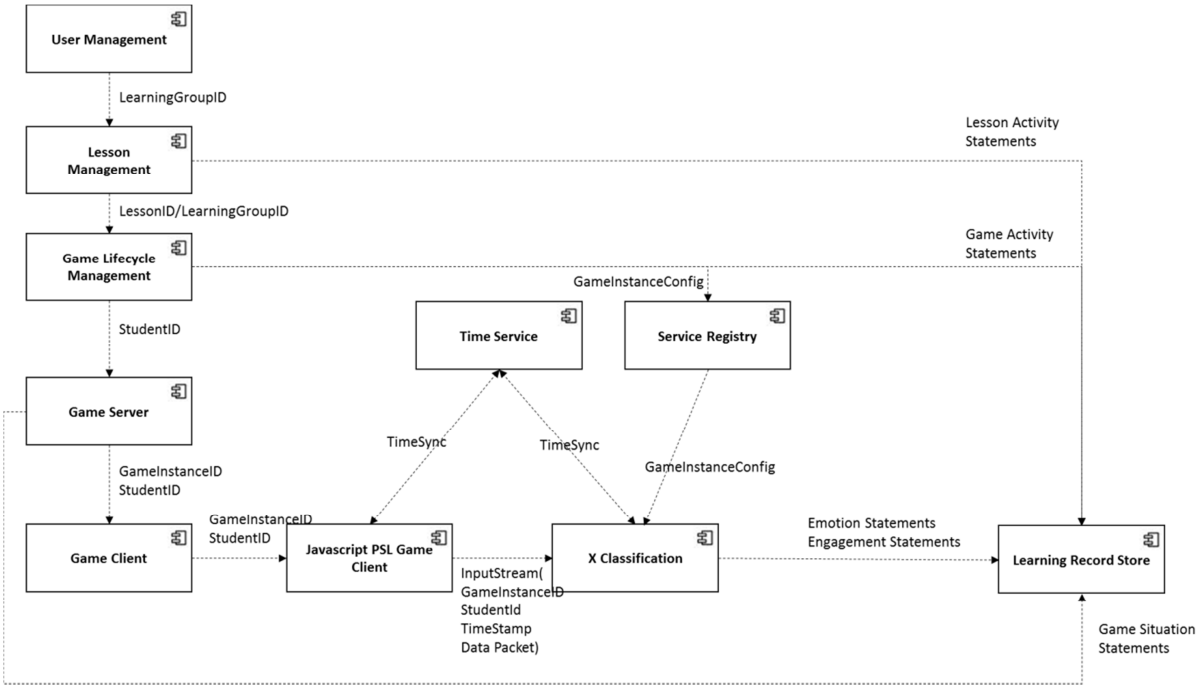

Figure 16: Data processing pipeline architecture

Each stream consists of frames of different sizes and frequencies depending on the observation, feature of interests and measurement. Each stream is identified uniquely by a Student ID and Game Instance ID. All frames are time stamped using a synchronised clock. The frame rates for different modalities are described in Table 30.

Observation	Input Frame	Output Frame
Face	Probabilities, 60fps, few bytes	Same as input

Body	Skeleton data, 60 fps, under 1k	Same as input
Voice	PCM audio, 10fps, 4k per frame	Valance arousal, 10 Hz

Table 30: Input and Output rates and frame sizes per observation modality

The classifier service model is designed to be stateless allowing for streams to be processed from any student interacting with any game instance, and allows classification service instances to be horizontally scaled based on demand. All classification services are synchronised with Platform time using the Timing service.

Classification data processing pipelines are high performance and resource intensive operations that needs to be managed appropriately considering the expected demands and responses times for near to real-time adaptation of games.

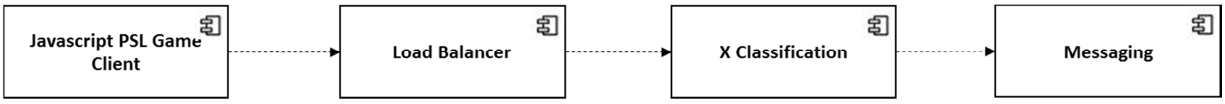

Figure 17: Classification performance management

The platform decouples input streams from classifiers using a load balancer to allow for classifier instances assigned to streams on demand and providing increased platform flexibility. The use of a load balancer allows for a common approach to stream buffering and scaling. Output streams are decoupled from downstream services using the Messaging component again allowing buffering to be implemented in a common way.

5.3.7.1 Facial Expression Classification Component

Facial expressions are considered a strong indicator of a person's emotional state. The platform uses vision-based automated frameworks for tracking and analysing activity of muscles and muscle groups, as defined by Ekman's Facial Action Coding System (FACS)[14].

The component utilizes a single, High-Definition (HD) webcam mounted on top of the player's monitor, in order to properly extract and analyse user facial activity in real time. Each frame is processed by a face detector (based on the Viola-Jones object detection algorithm[15] implemented in OpenCV[16]) and a facial feature extraction algorithm, as depicted in Figure 18. In cases where face detection is difficult, extraction of the face image colour histogram is used.

Facial expressions are described by Action Units (AUs) in accordance with FACS. AUs emotional states are described by combining AUs. AUs are defined and state changes observed using an Active Shape Model (ASM). ASMs are statistical models of the shape of faces which iteratively deform to fit to an input camera frame[17]. To identify displacements of facial features even at a micro-expression level the ASM employ a large number of features.

Tracking the movement of these facial feature points enables expression analysis to identify the activation of specific AUs and thus produce a measure of the player's real time emotional state.

[14] Ekman, Paul, and Wallace V. Friesen. "Facial action coding system." (1977).
[15] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proc. of the 2001 IEEE Computer Society Conference on* (Vol. 1, pp. I-511). IEEE.
[16] http://opencv.org/
[17] Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995). Active shape models-their training and application. *Computer vision and image understanding*, *61*(1), 38-59.

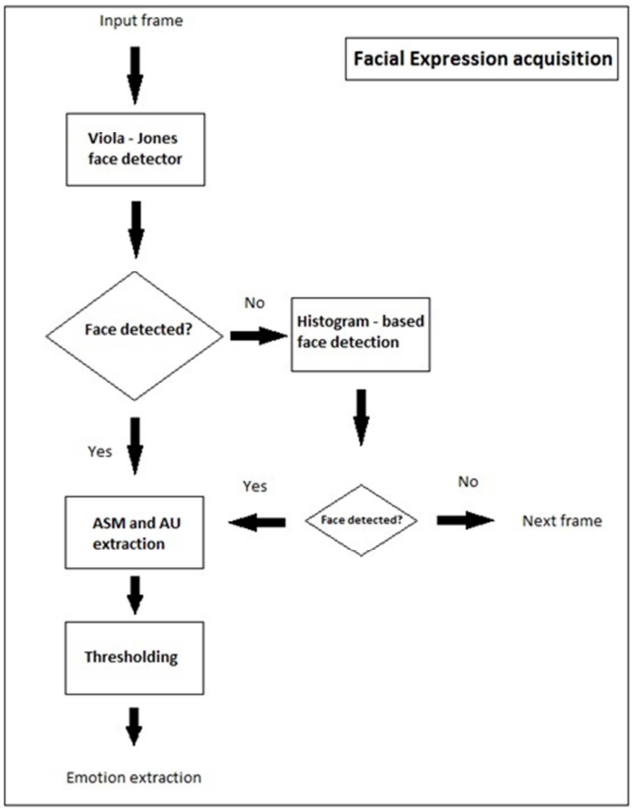

Figure 18: Overview of the facial expression acquisition component.

5.3.7.2 Eye-Gaze Tracking Component

Eye-gaze as a form of human-computer interface holds great promise for studying players' interaction with the game environment. Eye gaze data has been reported to provide information on emotional state[18], and may as well serve as an estimate for player engagement[19].

The eye gaze tracker operates alongside facial expression classification to determine gaze location on the screen, gaze distance, blinking and pupil diameter. Calibrated as well as un-calibrated gaze tracking frameworks can be deployed for unobtrusive integration with the gaming platform. Figure 19 provides an overview of data processing pipeline:

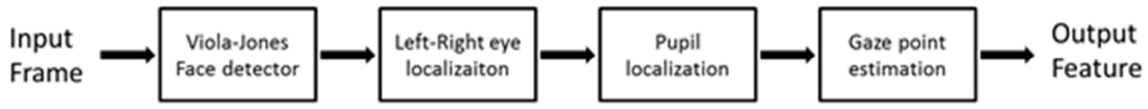

Figure 19: Overview of eye-gaze point estimation

5.3.7.3 Audio Classification Component

Voice classification uses sampled time varying signal and represent it in terms a number of useful features. The features employed are based on evidence from the literature[20] as to features which are

[18] Soleymani, M., Lichtenauer, J., Pun, T., & Pantic, M. (2012). A multimodal database for affect recognition and implicit tagging. *Affective Computing, IEEE Transactions on*, *3*(1), 42-55.

[19] Nakano, Y. I., & Ishii, R. (2010, February). Estimating user's engagement from eye-gaze behaviours in human-agent conversations. In *Proc. of the 15th Int. Conf. on Intelligent user interfaces* (pp. 139-148). ACM.

shown to provide efficacy in emotion detection. The incoming speech signal is captured using microphone sensors on client devices. The signal is windowed and features are computed within a time window. A window is needed in order to perform frequency domain analysis. The features extracted are in one of three classes: pitch, power spectrum, and formants (distinctive frequencies in signal). The initial extraction is used to derive additional features. Using pitch as an example the pitch feature can show the track of the fundamental frequency over time. In addition it is possible to compute the mean, variance, range, and first order differences. All features collected this way are augmented into a single feature vector which is passed onward to the fusion system.

5.3.8 Learning Analytics Subsystem

Acquiring knowledge of player's prosocial behaviours during game play necessary to understand competence in relation to desired prosocial skill, points of interest within game play situations requiring feedback and discussion and ways the game can be adapted to best suit individual needs.

5.3.8.1 Learning Record Store Component

The platform maintains a student and player profile in a Learning Record Store (LRS) responsible for storing game activity lifecycle events, prosocial skill measurements, emotional/engagement affect and other offline profile data.

Student profile data is analysed by social learning analytics to automatically assessment prosocial skills for teachers and game adaptation. The protocols for event reporting must be browser independent and standards (i.e. HTML5 and Javascript) based to ensure portability and interoperability. The WebSocket protocol is used for game interaction monitoring to provide near to real time monitoring channels for game adaptation or when recommendations to teachers for intervention (e.g. stopping the game). A Prosocial Learning Specification language has been defined building on the Experience API (xAPI) specification that is designed to track informal learning, social learning, and real world experiences. xAPI is highly applicable for performance assessment of learning activities within serious games and provides a flexible and standard-based solution for games developers and learning. The Prosocial Learning Specification language is generalised to enable reusability within many games and implement extensibility to support addition of new skills.

The Learning Record Store offers an xAPI endpoint against which other components can send xAPI statements about student performance. These statements are stored associated with all the complementary information around them (e.g. the player Id, the timestamp, the game Id). The LRS supports the possibility of performing queries on the stored data. A set of built-in queries are identified as able to assess prosocial skills that can be accessed by the Student Performance Dashboard (SPD).

The LRS also supports business rules <Event, Condition, Action> that can be used to activate a notification modality, such that the assessment can be pushed to external consumer (i.e. the Adaptation Manager and the Recommendation Service). This LRS also supports the definition and storing of additional custom queries and Event/Condition/Action rules that operates on the data.

The naming conventions for xAPI student performance statements are shown in Table 31

Resource	CreatedBy	Convention	Scope

[20] F. Dellaert, T. Polzin, and A. Waibel. Recognizing emotion in speech. In Proceedings of fourth international conference on spoken language processing, volume 3, pages 1970-1973, 1996.

Student performance measurement	Game Instance	A UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) according to the xAPI statement specification *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	Learning Provider
Student emotion and engagement measurement	Classifier Instance	A UUID (see RFC 4122 for requirements, and the UUID must be in standard string form) according to the xAPI statement specification *e.g. 82b604dd-de2d-4113-a1b6-bff0ebfcbe8b*	Learning Provider

Table 31: Naming convention for xAPI performance statements

5.3.8.2 Student Performance Dashboard Component

The Student Performance Dashboard (SPD) component interprets student profile data to feedback and recommendations to the teacher. The analytics are set of functions defined and applied to the student profile according to a skill ranking and performance model. For example, in a cooperative game where multiple players are learning skills for collaboration such as how to help others, a game defines function that measures the level of cooperation through the decisions made within the game and reports this to the LRS. These decisions can then be correlate game with temporal emotional affect measurements to gain an overall view on performance and affect for students.

The SPD automates the monitoring, analysis and visualisation of player behaviour, to offer teachers additional insight into student performance that is difficult to observe through traditional teaching methods. This allows Teachers to provide feedback and reflection on learning experiences to individual learning needs in accordance with the pedagogical needs. SPD provides visualisation of player performance data including general game statistics along with temporal views of prosocial skills and emotional affect during games that can be reviewed and replayed with students.

The SPD allows Teachers to define conditional triggers associated with observations and game interaction data for both individuals and groups of players to alert at points of interest, such as, one player is angry and at the same time another player makes an anti-social move, or one player is bored and disengaged from gameplay. The platform recognises that triggers are often context dependent and it requires teaching practitioners to determine the appropriate conditions when intervention is required and how such intervention should be implemented. For example, if a player becomes angry during game play it is for the teacher to decide if and when the game should be stopped and how feedback should be given, either as a group or individually.

At the same time, conditions can be used to trigger a signal to the Adaptation Manager recommending a change in the game. For example, if the collective level of cooperativeness in a cooperative game has been high yet the group did not win due to chance factors and emotion observation measures widespread frustration, then an alternative reward may be desirable. The analytics will provide an abstract trigger event (e.g. “recommend to deliver addition reward”) to the adaptation engine and ultimately to the game to implement within the specific game script.

5.3.8.3 Adaptation Manager Component

The Adaptation Manager (AM) provides personalization capabilities to enhance the process of instructing prosocial behaviour to students via gamification. Personalization provides simple configuration for individual needs allows the effective transmission and acquisition of knowledge for

individualised way of learning. The role of a teacher is to try and learn the individual preferences of students, and make students' learning as effective as possible. Therefore, considering the benefits gained from both expressions of personalization, the development of an adaptation manager that matches the preferences of players with the optimised game conditions (realising teaching conditions), is of great importance for the effectiveness of learning.

The AM is divided into two parts, namely the offline and the online adaptation mechanisms. The distinction concerns whether the processing takes place during gameplay or loading phase of the game. These two mechanisms aim to personalize the games towards maximizing the players' prosocial behaviour. Each mechanism processes different information about the player and concerns different types of factors affecting prosociality. Offline adaptation is based on historical player information that concerns their in-game performance, as well as, specific constraints of the Prosocial Learning Objectives (PLOs). While online adaptation considers real-time player data concerning player's engagement estimation during specific time intervals in the game.

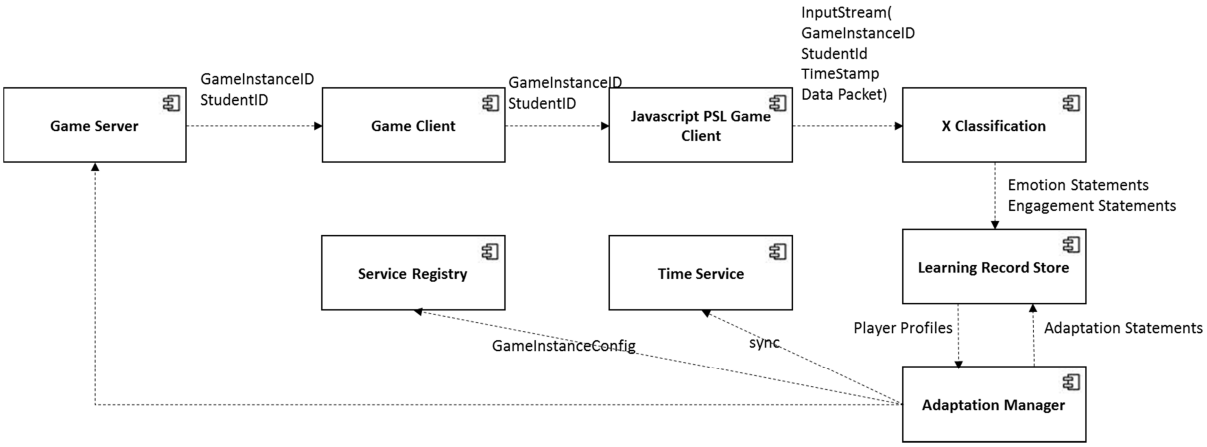

Figure 20: Adaptation architecture components

The components related to the Adaptation Manager are shown in Figure 20. Each Game Instance has an AM Instance responsible for real-time adaptation. When a Game Instance is initiated an AM is initialised by loading a player profile data for the set of students in the Learning Group from the LRS along with related Prosocial Learning Objectives defined by the teacher for the lesson. Throughout gameplay, the AM periodically synchronises prosocial state for the learning group from the LRS. The prosocial state information includes prosocial skills used and any player affect observations. The AM uses this information to calculate skill rankings for individual students in relation to the group both in totality and within the specific game situations played.

The ranking scores are used to calculate student performance and to make recommendations for adaptation. Each game offers game situations with points of variability classified according to the Prosocial Learn vocabulary for adaptation. If the AM determines that the Student would benefit from adaptation the AM recommends a change by asserting an adaptation statement to the game server via the messaging component. Any adaptation statements are also stored within the LRS.

5.3.9 Deployment Management Subsystem

The deployment management subsystem provides the functionality to deploy platform and game services on demand on a Virtualized Infrastructure. The deployment manager does not consider

what is necessary to deploy game clients and devices within school environments. Recommendations on these requirements are included within the Learning Unit metadata.

The platform aims to support a variety of game delivery models with the initial target being browser based online games.

Delivery Model	Description
Online	Games are fully operational remotely (in the cloud) through the Internet. Games are played through the common browser. Game servers and other servers are fully governed by the PSL platform. The advantage of this models is minimal deployment requirements for Learning Providers and device agnostic if games are developed using open standards. Any devices (tablets, mobiles, PC) any platform and Operating System: Windows (Microsoft), Android (Google), Mac iOS (Apple) can be supported
Downloadable	Games are fully downloaded onto the player device such as a desktop. In this case, the game is not agnostic to the device and would be complied for specific operating systems, and these programs would access Platform services deployed within the cloud directly.
Hybrid	Games clients are downloaded and installed on a specific device but access a game server hosted within the Platform. Also here, the game is not agnostic to the device.

Table 32: Game delivery models

The platform supports an efficient gaming applications deployment and management using PaaS model. Platform and game services are packaged as Docker images that are deployed in Docker containers managed by the Platform. Docker is used to provide standardised software units that load quickly on demand.

Using Docker and ANSIBLE together provides the automated provisioning and orchestration of containers.

The platform monitors Docker resource metrics (e.g. CPU, memory, I/O, and network resources) that are provided to the RM component to allow the platform operator to proactively monitor performance against platform user access agreements. The naming convention for service instances are shown in Table 33

Resource	CreatedBy	Convention	Scope
Platform Services	Deployment Management	URL <platform-namespace>\service a unique referenceable endpoint URL for all platform services such as Marketplace Service, Learning Record Store xAPI Service, Classification Services *e.g.* *www.prosociallearn.eu\marketplace*	Global

Table 33: Naming convention for service instances

5.3.9.1 Cloud Management Subsystem

PSL platform builds on Cloud Foundry (CF). CF is an open source cloud computing platform as a service (PaaS) that provides a virtualisation environment that allows the platform provider to deploy, manage and scale cloud-based applications (games in the context of PSL) quickly, and easily.

CF allows governing software artifacts deployed on it, supporting the full lifecycle of games from the initial development, through all testing stages, to deployment. It is therefore well-suited to the continuous delivery strategy. Users have access to one or more spaces, which typically correspond to a lifecycle stage.

In the context of the ProsocialLearn project, a local instance of CF has been installed and managed.

A range of standardized infrastructure metrics will be monitored to allow a Platform provider to manage Quality of Service.

5.3.9.2 Security Subsystem

The security architecture brings together people, processes, and tools to protect data, service and infrastructure assets owned by platform stakeholders. The architecture is driven by regulatory requirements for data protection and privacy, and is structured to efficiently and effectively support platform business processes.

5.3.9.3 Security Domains and Trust

The security architecture defines a set of distinct security domains that delimitate physical and logical units where a single and homogeneous security policy is valid and applied. Each domain has security policy that controls the behaviour of the security services being provided.

Secure interaction between domains must be in accordance with agreed security policy governing this interaction. Agreed policies must be implemented within each individual domain.

ProsocialLearn operates in a multi-domain environment where logical domains span multiple physical domains.

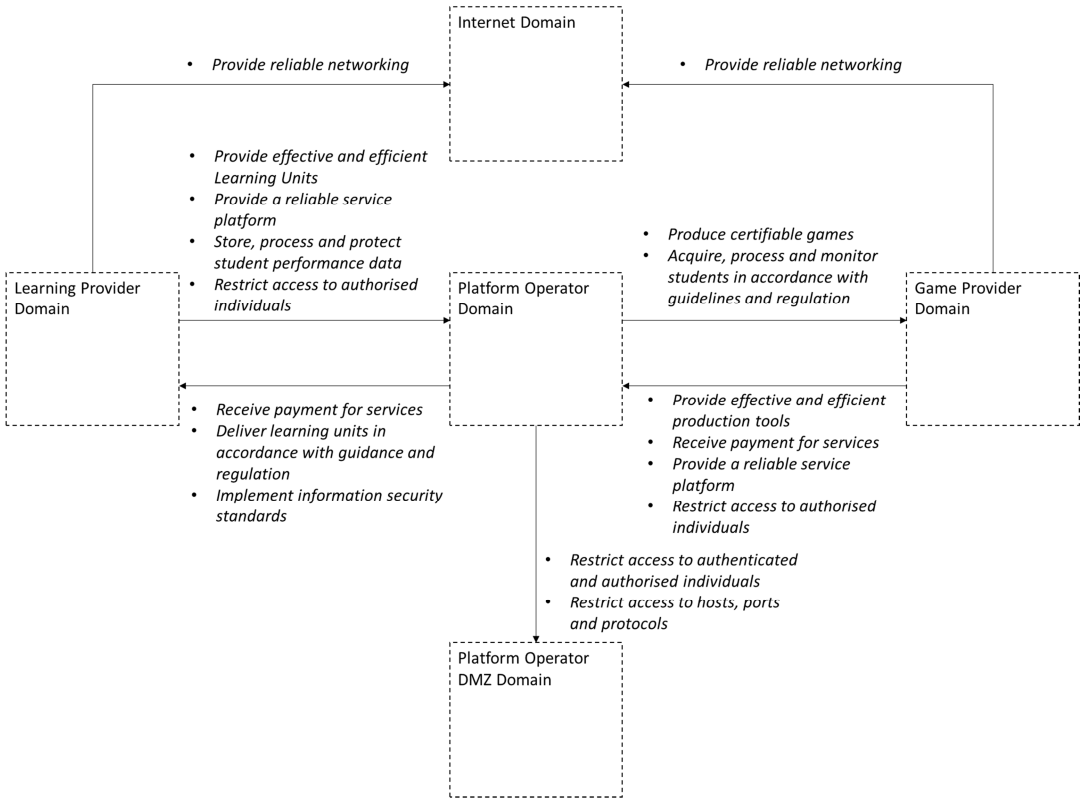

Figure 21: Security and trust domain model

The physical security domains (i.e. sites, platform and network) and their respective trust relationships are shown in Figure 21. Each domain must implement security policy to protect their respective assets in accordance with the trusted relationships:

- Learning Provider Domain: A site that delivers prosocial learning to children typically in an educational context such as a school. The Learning Provider is the data controller for subjects within the domain (i.e. children and teachers) and responsible for ensuring data protection including ensuring processing of data conducted by 3rd parties such as the PSL Platform is performed in accordance with relevant regulations (e.g. EU Directive 95/46 and nation state implementations). The Learning Provider must operating an IT infrastructure to manage and deliver learning activities including servers, desktop machines, laptop machines, tablets connected by LAN (Ethernet and Wi-Fi) with Internet connectivity. The infrastructure must be protected in accordance with international standards for information security (e.g. ISO/IEC 27001)
- Game Provider Domain: A site that develops prosocial games typically in a commercial context such as an SME. The SME may be a virtual organisation where developers are collaborating across multiple sites. Early game development must use anonymised test data sets to verify and validate function and performance. User testing in small trials and open beta programmes prior to publication of games to the marketplace must be apply all relevant security policy for production use as required by Data Controllers
- PSL Demilitarized Zone (DMZ) Domain: A physical or logical subnetwork that contains and exposes the Platform externally facing services to the untrusted Internet. All access to Platform services from external domains must only be through equipment with the DMZ.

Hosts within the DMZ must only have limited connectivity to hosts in the internal network and external network restrictions must also be applied.

- Platform Operator Domain: A physical or logical subnetwork that hosts all platform services and data assets in a cloud environment. The Platform Operator is a Data Processor responsible for processing personal information on behalf of a Learning Provider. The Platform Operator must comply with all relevant data protection regulation and apply all information security standards for clouds (e.g. ISO/IEC 27017)
- Internet Domain: An untrusted network operated by a ISPs and Network Operators to provide global connectivity and communication. Stakeholders within this domain have no responsibility for policies in relation to over-the-top applications running on their network and as such the Internet Domain is untrusted.

5.3.9.4 Identity Management

Identify management is concerned with controlling information that is used to identify platform users and services for the purpose of allowing access control permissions to be assigned and evaluated against this identify. User registration and identifiers are described in Section 5.3.2 **¡Error! No se encuentra el origen de la referencia.**.

The platform provides single sign through one credential for all services offered by the platform covering collaboration tools, management services and games themselves.

The platform does not currently support OAuth 2.0 as the requirement for users to be able to use accounts on different service providers and to selectively provide access to platform services contravene current registration and security policies.

5.3.9.5 Access Control

The platform restricts access to resources to only users that are authenticated and authorised in accordance with well-defined business processes and security policy consequences.

Access control policies define rules that specify access privileges to protected resources. The Platform possesses resources and services need to be protected, managed and monitored. The access control policies control the access permissions and usage of these resources by defining when and how a user can perform an action on a given resource.

A policy statement consists of *rules, subjects and conditions* that must be satisfied for access to be permitted. Table 34 describes the structure of access control policy statements.

Policy Element	Description	Example
Rule	A rule contains a service type and one or more actions with an appropriate value that, in effect, defines the intent of the policy. The service type defines the policy-enabled resource that is being protected. An action is the operation that can be performed on the resource; examples of web server actions are POST or GET. A value defines the permission for the action, for example, Allow or Deny.	An allowable action for the Lesson Management subsystem might be to change a the lesson start time

Subject	A subject defines the user or collection of users (for instance, a group or those who possess a specific role) that the policy affects. The general rule for subjects is that the policy would apply only if the user is a member of at least one subject in the policy.	An *Authenticated Teacher* with a valid SSOToken, Game Clients with a valid SSOToken is a member IF the Distinguished Name (DN) of any principal contained in the SSOToken matches any value of this subject, an LDAP Group representing a Learning Group
Condition	A condition constraints the policy	An authenticated student can only access a Lesson within the time it is scheduled

Table 34: Access control policy structure

Access control policies are created and updated as a consequence of platform business processes.

When a user takes an action on a service or resource the action results in policy updates. For example, when a Teacher creates a Lesson between 10am-12pm on Monday morning for a Learning Group: Class A, policy statements are created for the Lesson that allow Class A to access between the specified times.

Each platform subsystem defines business processes that access platform resources such as the marketplace, lessons and games. A preliminary list of policy statements for the platform is given below.

Rule	**Subject**	**Condition**
Marketplace (publish)	Game Provider	Active game provider account exists
Marketplace (search)	Game Provider, Budget Holder, Teacher	Active game provider or learning provider account exists
Learning Resource Management	Game Provider	Active game provider account exists
User Management (Teachers)	Budget Holder, Learning Provider Admin	Active learning provider account exists
User Management (Students)	Teachers, Learning Provider Admin	Active learning provider account exists
User Management (Game Developers)	Game Provider	Active game provider account exists
Lesson Management (Create, Update, Delete)	Teacher	Active learning provider account exists
Lesson Management (Retrieve)	Teacher, Student	Active learning provider account exists
Learning Analytics (Create, Update,	Game Client, Game Server	Active game instance exists

Retrieve)		
Learning Analytics (Create, Update, Retrieve, Delete)	Teacher	Active learning provider account exists
Learning Analytics (Create, Update, Retrieve, Delete)	Platform Client	None
Support Services (Create, Update, Delete)	Platform Client	None
Support Services (Retrieve)	Game Client, Game Server, Platform Client	
Cloud Management (Create, Retrieve, Update, Delete)	Platform Client	None

Table 35: Summary of key access control policy statements

5.3.9.6 Border control

All domains must operate a firewall to restrict incoming and outgoing connections according to their security policy. Typically a Learning Provider domain will be restricted to outgoing initiated connections on Port 80 (HTTP) or Port 443 (HTTPS). The Prosocial API is designed based to build on the HTTPS protocol using http, web sockets and rtc.

5.3.9.6.1 Security Gateway

The Platform Provider operates a centralised Security Gateway providing a reverse proxy policy enforcement point for all platform services. All external access to the platform enters through the security gateway.

The gateway enforces transport layer security, authentication and authorisation using the AuthN/AuthZ policy decision point. The gateway provides URL translation from external URLs to internal services hosted with Platform Provider's internal trusted domain.

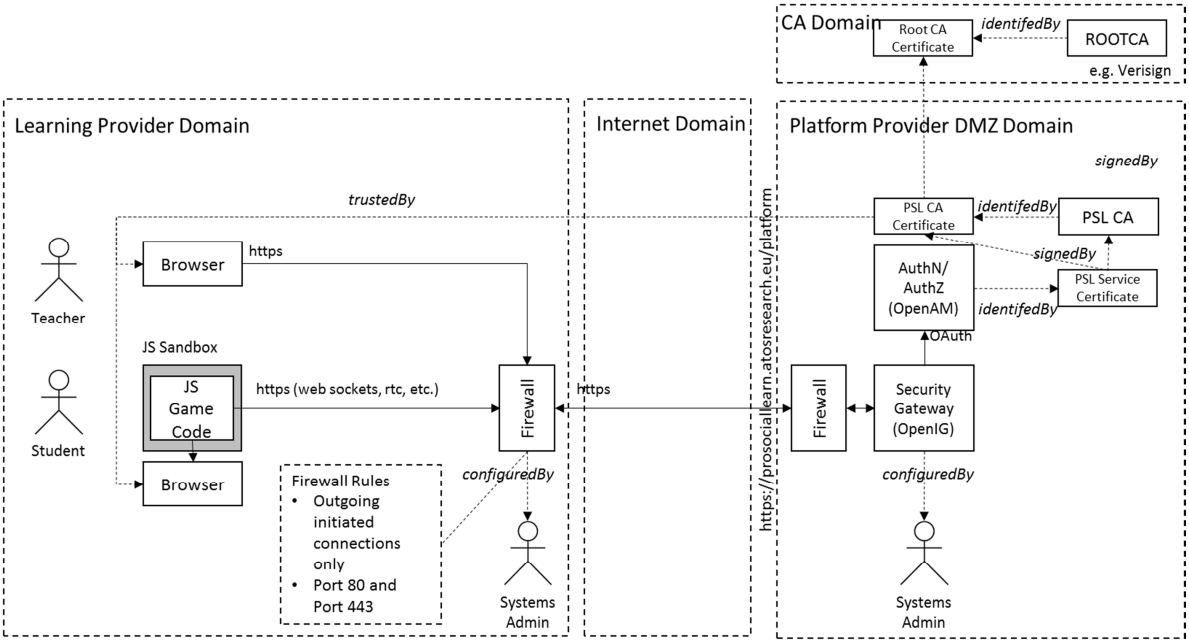

Figure 22: Border Control Architecture

5.3.9.7 Transport Layer Security

Interaction between Learning Provider/Game Provider Domains and the Platform Domain is over an untrusted network, the Internet. To secure interaction over this channel uses protocols based on HTTPS using strong encryption.

The root of trust for the communication is the Prosocial Learn Certificate Authority CA that asserts identify of host machines deployed on the platform. Learning Provider users and Game Provider users must add the public ProsocialLearn CA Certificate to the trust store (i.e. in the browser) of client machines wanting to access the platform so that the signature of the platform machines can be verified when setting up a secure connection.

5.3.9.8 Javascript Sandboxing and API Keys

Many games execute within a browser using Javascript and WebGL. Browsers execute code within a sandbox with constraints on access to capabilities of the physical device and restricted access to only hosts serving up the code via the Security Gateway.

Some 3rd party platform services use API keys to authenticate and authorise client applications. The platform does not need such client applications to be identified but must provider a key to use the API. The platform therefore manages API keys within the Security Gateway, inserting the required token and password information into the session context when such service is invoked. This is required for client side Javascript code where API keys cannot be securely in code or sessions as the code is visible and session state can easily be viewed using debugging tools.

5.4 Scalability

Platform scalability refers to how the system gets bigger (and smaller) in response to demand in terms of users, sessions, transactions, data processing, etc. The platform must scale with minimal effort as the demand for prosocial gaming within the education sector grows from 100's to 100's of

thousands of students. The platform size is driven by the different types of data processing, data storage and related performance requirements.

- Transactional Subsystems: User Management, Marketplace, Learning Resource Management, Lesson Management, Cloud Management and Support Services
- Real-time Stream Processing Subsystems: Player Affect Fusion, Learning Analytics

Transaction scalability is well understood and presents no architectural challenges. The architecture builds on a cloud infrastructure that offers the fundamental capability to horizontally scale platform services, and when used in conjunction with load balancing can optimally distribute workloads across multiple computing servers, ensure high availability and reliability by sending requests only to servers that are online, and provides the flexibility to upscale or downscale servers on demand.

Real-time stream processing is however complex driving the need for big data capacity (bandwidth, memory, storage) and performance considering the quantity of player affect observations. For example, Table 36 shows a scalability analysis for a Learning Provider under different scenarios of concurrent use for students participating in lessons observed across four modalities (audio, face, body and PSL-xAPI skill assertions).

Scalability is concerned with total usage of consumable resources (e.g. disk storage) and peak usage of reusable resources (e.g. memory and network bandwidth).

- How many game play hrs can be performed at a level of concurrency and storage constraint?
- How many average lessons can be conducted with a given storage capacity?
- What is the peak memory usage and network bandwidth at different levels of concurrency?

	Scale 1	Scale 2	Scale 3	Scale 4	Scale 5	Description
Number of concurrent students playing	10	25	100	1000	100000	Integer
Number of game play seconds	3600	3600	3600	3600	3600	Seconds
Number of observation modalities	4	4	4	4	4	Audio, Face, Body, In Game
Average class size	25	25	25	25	25	Integer
Frequency of observation	1	1	1	1	1	hz
Average Valance Arousal Classifier						
Sample Window Size (V+A)	8	8	8	8	8	kbytes
Windows per second	5	5	5	5	10	Integer
Number of window buffers	5	5	5	5	5	Integer
Data stream memory	200	200	200	200	400	kbytes
Other memory	200	200	200	200	200	kbytes
Total memory usage per stream per second	400	400	400	400	600	kbytes
Total memory	0.016	0.04	0.16	1.6	240	kbytes
Classfier network bandwidth	3.2	8	32	320	64000	Mbps
Learning Record Store						
PSL-xAPI average statement size	300	300	300	300	300	bytes
Number of PSL-xAPI statements created	144000	360000	1440000	14400000	1440000000	Integer
Number of PSL-xAPI Statements per second	40	100	400	4000	400000	Integer
Learning Record Store Storage Capacity Constraint	500	1000	1000	2000	10000	GByte
Number of WT Compressed Statement Capacity	706783893	1413567786	1413567786	2827135572	14135677860	Integer
Maximum gameplay hrs at level of concurrency	4908.221479	3926.577183	981.6442958	196.3288592	9.816442958	Integer
PSL-xAPI network bandwidth	0.096	0.24	0.96	9.6	960	Mbps
Memory needed per 500G storage	16	16	16	16	16	Gbyte
LRS Memory per 500G storage	16	32	32	64	320	GByte
Totals						
Network bandwidth	3.296	8.24	32.96	329.6	64960	Mbps
Memory	19.2	40	64	384	64320	GBytes

Table 36: Platform scalability analysis for different demand scenarios

The analysis shows that for an average Learning Group size of 25 students an LRS with 1TByte storage can store 3926 lessons of 1 hr and would require a network of 8.24Mbps and 40Gbytes of RAM. If a Learning Provider has 500 students with 20 Learning Groups, delivering 1 prosocial lesson to each Learning Group per week then 1040 lessons would be delivered per annum. In which case, 1 Tbyte of storage would be sufficient for 3.77 years of usage.

6 Conclusions

This deliverable describes the 2nd version of the ProsocialLearn system requirements and architecture. The document focuses on providing a coherent and consistent description of the capabilities of the platform necessary for end users to understand the requirements and for developers to implement and integrate the final system.

Although the document is the final official version of the systems requirements and architecture the information contained within this version has now been transferred to the Developers Guide on the Prosocial Learn Master Git Hub project. This augments the detailed developer's documentation with high level information and allows for new game developers to fully understand the capabilities of the prosocial learn platform in context. The benefits of Git Hub is that the document becomes part of the platform release and is incorporated into version control.