

توصيف النظام

اسم الشركة:

شركة البرمجيات "SAVVY"

نظرة عامة:

شركة البرمجيات "SAVVY" تقدم حلولاً تقنية مبتكرة في مجال تطوير البرمجيات وإدارة المشاريع. تهدف الشركة إلى تزويد عملائها بحلول متكاملة في تطوير البرمجيات، حيث تقدم خدمات متميزة لعملائها من خلال مشاريع مختلفة ومتنوعة، سواء كانت في مجال البرمجيات أو الإدارة.

الهيكل التنظيمي:

تتألف الشركة من عدة أقسام رئيسية تشمل:

1. الموظفون (Employee):

- الموظفون في الشركة هم العنصر الأساسي في نجاح الأعمال. كل موظف في الشركة يتمتع بكفاءة عالية وهو متخصص في مجاله.
- تتم إدارة الموظفين وفقاً للأدوار المختلفة مثل (إداري، مدير، موظف) ويعملون على مختلف المشاريع.

2. العملاء (Client):

- تعتبر الشركة العملاء هم أساس وجودها، وهي تقوم بإدارة المعلومات المتعلقة بالعملاء مثل الاسم، العنوان، البريد الإلكتروني، الهاتف، وكذلك تاريخ الزيارة الأولى.

3. المشاريع (Project):

- تعمل الشركة على تنفيذ مشاريع تقنية متنوعة تتطلب فريقاً مخصصاً. يتم تسجيل تفاصيل كل مشروع بما في ذلك العنوان، الوصف، التواريخ، السعر، وتحديد ما إذا كان المشروع قد انتهى أو تم إلغاؤه.

4. الفواتير والمدفوعات (Payments and Receipts):

- تتعامل الشركة مع المدفوعات المترتبة على المشاريع المختلفة من خلال مجموعة من الآليات مثل الدفع النقدي، الدفع بواسطة بطاقة الائتمان، أو التحويل البنكي. كما يتم إصدار الفواتير المتعلقة بهذه المدفوعات.

5. المهام والتقارير (Tasks and Reports):

- يتم تنظيم العمل في الشركة عبر مجموعة من المهام التي يتعين على الموظفين إتمامها ضمن مشاريع معينة، إضافة إلى إعداد تقارير عن تقدم الأعمال.

وظائف المشروع الشاملة:

العمليات على الموظفين:

1. عرض الموظفين.
2. إضافة موظف.
3. حذف موظف.
4. تعديل بيانات موظف.
5. تسجيل دخول الموظف.

العمليات على المقبوضات:

1. عرض مقبوضات الشهر الحالي.
2. عرض مقبوضات شهر معين.
3. حذف مقبوض.
4. تعديل مقبوض.
5. إضافة قبض جديد.
6. عرض مجموع مقبوضات السنة الحالية.
7. عرض مقبوضات إحدى السنوات.

العمليات على العملاء:

1. إضافة عميل جديد.
2. حذف عميل ما.
3. تعديل بيانات أحد العملاء.
4. عرض جميع العملاء.
5. عرض عملاء أحد السنين.

العمليات على المشاريع:

1. إضافة مشروع جديد.
2. تعديل على المشروع.
3. حذف أحد المشاريع.
4. إلغاء مشروع.
5. إنهاء مشروع.
6. عرض المشاريع مفلتر.
7. عرض مشاريع السنة الحالية.
8. عرض مشاريع أحد السنين.
9. عرض مجموع دخل كل المشاريع لأحد السنين.
10. عرض مجموع دخل مشاريع السنة الحالية.

العمليات على التقارير:

1. إضافة تقرير.
2. حذف تقرير.
3. تعديل تقرير.
4. عرض تقارير أحد الموظفين.
5. عرض جميع التقارير حسب الشهر والسنة.
6. عرض جميع تقارير السنة والشهر الحاليين.

العمليات على المهام:

1. إضافة مهمة.
2. حذف مهمة.
3. تعديل مهمة.
4. إنهاء مهمة.
5. عرض مهام أحد الموظفين.
6. عرض المهام حسب المشروع.
7. عرض مهام الشهر الحالي.

العمليات على سجل العمليات:

1. عرض العمليات التي قام بها كل موظف.
2. عرض عمليات الشهر الحالي.
3. حذف عملية.
4. إضافة عملية.
5. تعديل عملية.

العمليات على المدفوعات:

1. عرض جميع المدفوعات.
2. عرض مدفوعات السنة الحالية.
3. إضافة مدفوعات.
4. حذف مدفوعات.
5. تعديل بيانات المدفوعات.

وصف قاعدة البيانات

وصف قاعدة البيانات

1. جدول Employee (الموظف)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل موظف.
- name (VARCHAR(100)): اسم الموظف.
- hiredate (DATE): تاريخ تعيين الموظف.
- salary (DECIMAL(10,2)): راتب الموظف.
- permission (ENUM('admin', 'manager', 'staff')): صلاحية الموظف.
- phone (VARCHAR(20)): رقم الهاتف الخاص بالموظف.
- cv_file (TEXT): رابط السيرة الذاتية للموظف.
- type (VARCHAR(50)): نوع الموظف.
- last_login (DATETIME): آخر تسجيل دخول للموظف.

2. جدول Client (الزبون)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل زبون.
- name (VARCHAR(100)): اسم الزبون.
- address (TEXT): عنوان الزبون.
- email (VARCHAR(100)): البريد الإلكتروني للزبون.
- phone (VARCHAR(20)): رقم الهاتف الخاص بالزبون.
- first_visit_date (DATE): تاريخ أول زيارة للزبون.
- company_name (VARCHAR(100)): اسم الشركة الخاصة بالزبون.

3. جدول Project (المشروع)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل مشروع.
- title (VARCHAR(150)): عنوان المشروع.
- description (TEXT): وصف المشروع.
- begin_date (DATE): تاريخ بدء المشروع.
- end_date (DATE): تاريخ انتهاء المشروع.
- price (DECIMAL(10,2)): سعر المشروع.
- notes (TEXT): ملاحظات حول المشروع.
- is_finish (TINYINT(1)): حالة انتهاء المشروع (1 = منتهي، 0 = غير منتهي).
- is_canceled (TINYINT(1)): حالة إلغاء المشروع (1 = ملغي، 0 = غير ملغي).
- employee_manager_id (INT, FK): معرف الموظف المسؤول عن المشروع.
- client_id (INT, FK): معرف الزبون المرتبط بالمشروع.

4. جدول Task (المهمة)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل مهمة.
- employee_id (INT, FK): معرف الموظف المسؤول عن تنفيذ المهمة.
- start_date (DATE): تاريخ بدء المهمة.
- finish_date (DATE): تاريخ انتهاء المهمة.
- notes (TEXT): ملاحظات حول المهمة.
- type (VARCHAR(50)): نوع المهمة.
- level (ENUM('low', 'medium', 'high')): مستوى أولوية المهمة.
- project_id (INT, FK): معرف المشروع المرتبطة به المهمة.

5. جدول Receipt (عملية القبض)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل عملية قبض.
- receipt_no (VARCHAR(50)): رقم الإيصال.
- employee_id (INT, FK): معرف الموظف الذي استلم المبلغ.
- receipt_date (DATE): تاريخ عملية القبض.
- amount (DECIMAL(10,2)): المبلغ المستلم.
- receipt_note (TEXT): ملاحظات حول عملية القبض.

6. جدول Payments (المدفوعات)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل عملية دفع.
- client_id (INT, FK): معرف الزبون الذي قام بالدفع.
- project_id (INT, FK): معرف المشروع المرتبط بالدفع.
- amount (DECIMAL(10,2)): قيمة المبلغ المدفوع.
- payment_date (DATE): تاريخ الدفع.
- payment_method (ENUM('cash', 'credit_card', 'bank_transfer')): وسيلة الدفع.
- notes (TEXT): ملاحظات حول عملية الدفع.

7. جدول Logs (سجل العمليات)

- id (INT, المفتاح الأساسي، زيادة تلقائية): معرف فريد لكل سجل عملية.
- employee_id (INT, FK): معرف الموظف الذي قام بالإجراء.
- action (VARCHAR(255)): وصف الإجراء الذي تم تنفيذه.
- action_time (TIMESTAMP): وقت تنفيذ الإجراء.


```
CREATE TABLE `client` (
  `id` int(10) UNSIGNED NOT NULL,
  `name` varchar(100) NOT NULL,
  password_hash varchar(255) not null
  `address` text NOT NULL,
  `email` varchar(100) NOT NULL,
  `phone` varchar(20) NOT NULL,
  `first_visit_date` date NOT NULL,
  `company_name` varchar(100) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `employee` (
  `id` int(10) UNSIGNED NOT NULL,
  `name` varchar(100) NOT NULL,
  `hiredate` date NOT NULL,
  `salary` decimal(10,2) NOT NULL CHECK (`salary` >= 0),
  `permission` enum('admin','manager','staff') NOT NULL,
  `phone` varchar(20) NOT NULL,
  `cv_file` text DEFAULT NULL,
  `type` varchar(50) NOT NULL,
  `last_login` datetime DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `logs` (
  `id` int(10) UNSIGNED NOT NULL,
  `employee_id` int(10) UNSIGNED NOT NULL,
  `action` varchar(255) NOT NULL,
  `action_time` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `payments` (
  `id` int(10) UNSIGNED NOT NULL,
  `client_id` int(10) UNSIGNED NOT NULL,
  `project_id` int(10) UNSIGNED NOT NULL,
  `amount` decimal(10,2) NOT NULL CHECK (`amount` >= 0),
  `payment_date` date NOT NULL,
  `payment_method` enum('cash','credit_card','bank_transfer') NOT NULL,
  `notes` text DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `project` (
  `id` int(10) UNSIGNED NOT NULL,
  `title` varchar(150) NOT NULL,
  `description` text DEFAULT NULL,
```

```
`begin_date` date NOT NULL,  
`end_date` date NOT NULL,  
`price` decimal(10,2) NOT NULL CHECK (`price` >= 0),  
`notes` text DEFAULT NULL,  
`is_finish` tinyint(1) DEFAULT 0,  
`is_canceled` tinyint(1) DEFAULT 0,  
`employee_manager_id` int(10) UNSIGNED DEFAULT NULL,  
`client_id` int(10) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `receipt` (  
  `id` int(10) UNSIGNED NOT NULL,  
  `receipt_no` varchar(50) NOT NULL,  
  `employee_id` int(10) UNSIGNED NOT NULL,  
  `receipt_date` date NOT NULL,  
  `amount` decimal(10,2) NOT NULL CHECK (`amount` >= 0),  
  `receipt_note` text DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `report` (  
  `id` int(10) UNSIGNED NOT NULL,  
  `title` varchar(150) NOT NULL,  
  `description` text NOT NULL,  
  `date` date NOT NULL,  
  `employee_id` int(10) UNSIGNED NOT NULL,  
  `project_id` int(10) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE `task` (  
  `id` int(10) UNSIGNED NOT NULL,  
  `employee_id` int(10) UNSIGNED NOT NULL,  
  `start_date` date NOT NULL,  
  `finish_date` date NOT NULL,  
  `notes` text DEFAULT NULL,  
  `type` varchar(50) NOT NULL,  
  `level` enum('low','medium','high') NOT NULL,  
  `project_id` int(10) UNSIGNED NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
ALTER TABLE `client`  
  ADD PRIMARY KEY (`id`),  
  ADD UNIQUE KEY `email` (`email`),  
  ADD UNIQUE KEY `phone` (`phone`);
```

```
ALTER TABLE `employee`  
  ADD PRIMARY KEY (`id`),
```



```
ADD UNIQUE KEY `phone` (`phone`);
```

```
ALTER TABLE `logs`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `employee_id` (`employee_id`);
```

```
ALTER TABLE `payments`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `client_id` (`client_id`),  
  ADD KEY `project_id` (`project_id`);
```

```
ALTER TABLE `project`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `employee_manager_id` (`employee_manager_id`),  
  ADD KEY `client_id` (`client_id`);
```

```
ALTER TABLE `receipt`  
  ADD PRIMARY KEY (`id`),  
  ADD UNIQUE KEY `receipt_no` (`receipt_no`),  
  ADD KEY `employee_id` (`employee_id`);
```

```
ALTER TABLE `report`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `employee_id` (`employee_id`),  
  ADD KEY `project_id` (`project_id`);
```

```
ALTER TABLE `task`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `employee_id` (`employee_id`),  
  ADD KEY `project_id` (`project_id`);
```

```
ALTER TABLE `client`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;
```

```
ALTER TABLE `employee`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;
```

```
ALTER TABLE `logs`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;
```

```
ALTER TABLE `payments`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;
```

```
ALTER TABLE `project`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;
```

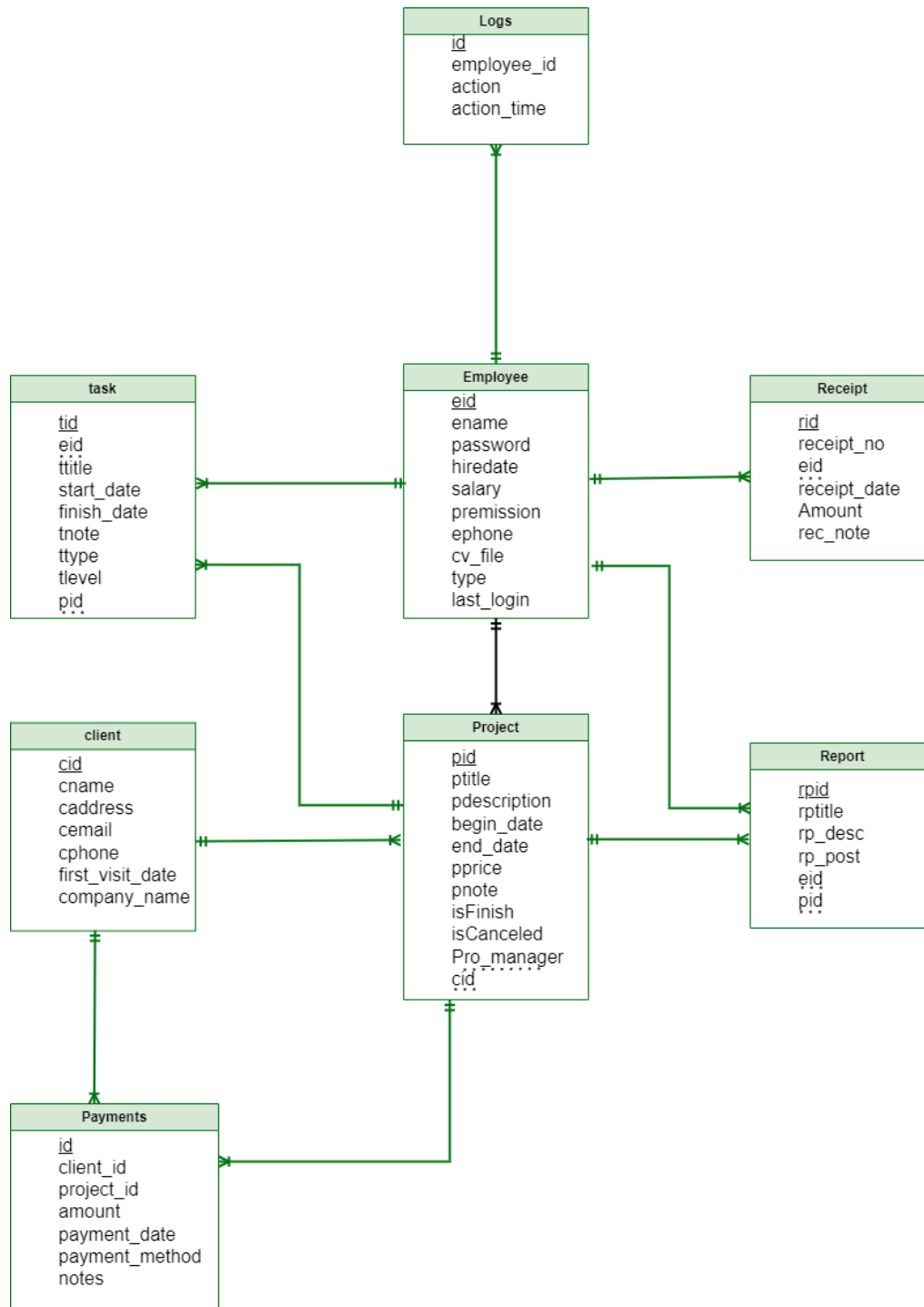


```
ALTER TABLE `receipt`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;  
  
ALTER TABLE `report`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;  
  
ALTER TABLE `task`  
  MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT;  
  
ALTER TABLE `logs`  
  ADD CONSTRAINT `logs_ibfk_1` FOREIGN KEY (`employee_id`) REFERENCES  
`employee` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;  
  
ALTER TABLE `payments`  
  ADD CONSTRAINT `payments_ibfk_1` FOREIGN KEY (`client_id`) REFERENCES  
`client` (`id`) ON DELETE CASCADE,  
  ADD CONSTRAINT `payments_ibfk_2` FOREIGN KEY (`project_id`) REFERENCES  
`project` (`id`) ON DELETE CASCADE;  
  
ALTER TABLE `project`  
  ADD CONSTRAINT `project_ibfk_1` FOREIGN KEY (`employee_manager_id`) REFERENCES  
`employee` (`id`) ON DELETE SET NULL,  
  ADD CONSTRAINT `project_ibfk_2` FOREIGN KEY (`client_id`) REFERENCES  
`client` (`id`) ON DELETE CASCADE;  
  
ALTER TABLE `receipt`  
  ADD CONSTRAINT `receipt_ibfk_1` FOREIGN KEY (`employee_id`) REFERENCES  
`employee` (`id`) ON DELETE CASCADE;  
  
ALTER TABLE `report`  
  ADD CONSTRAINT `report_ibfk_1` FOREIGN KEY (`employee_id`) REFERENCES  
`employee` (`id`) ON DELETE CASCADE,  
  ADD CONSTRAINT `report_ibfk_2` FOREIGN KEY (`project_id`) REFERENCES  
`project` (`id`) ON DELETE CASCADE;  
  
ALTER TABLE `task`  
  ADD CONSTRAINT `task_ibfk_1` FOREIGN KEY (`employee_id`) REFERENCES  
`employee` (`id`) ON DELETE CASCADE,  
  ADD CONSTRAINT `task_ibfk_2` FOREIGN KEY (`project_id`) REFERENCES  
`project` (`id`) ON DELETE CASCADE;
```



مخططات UML

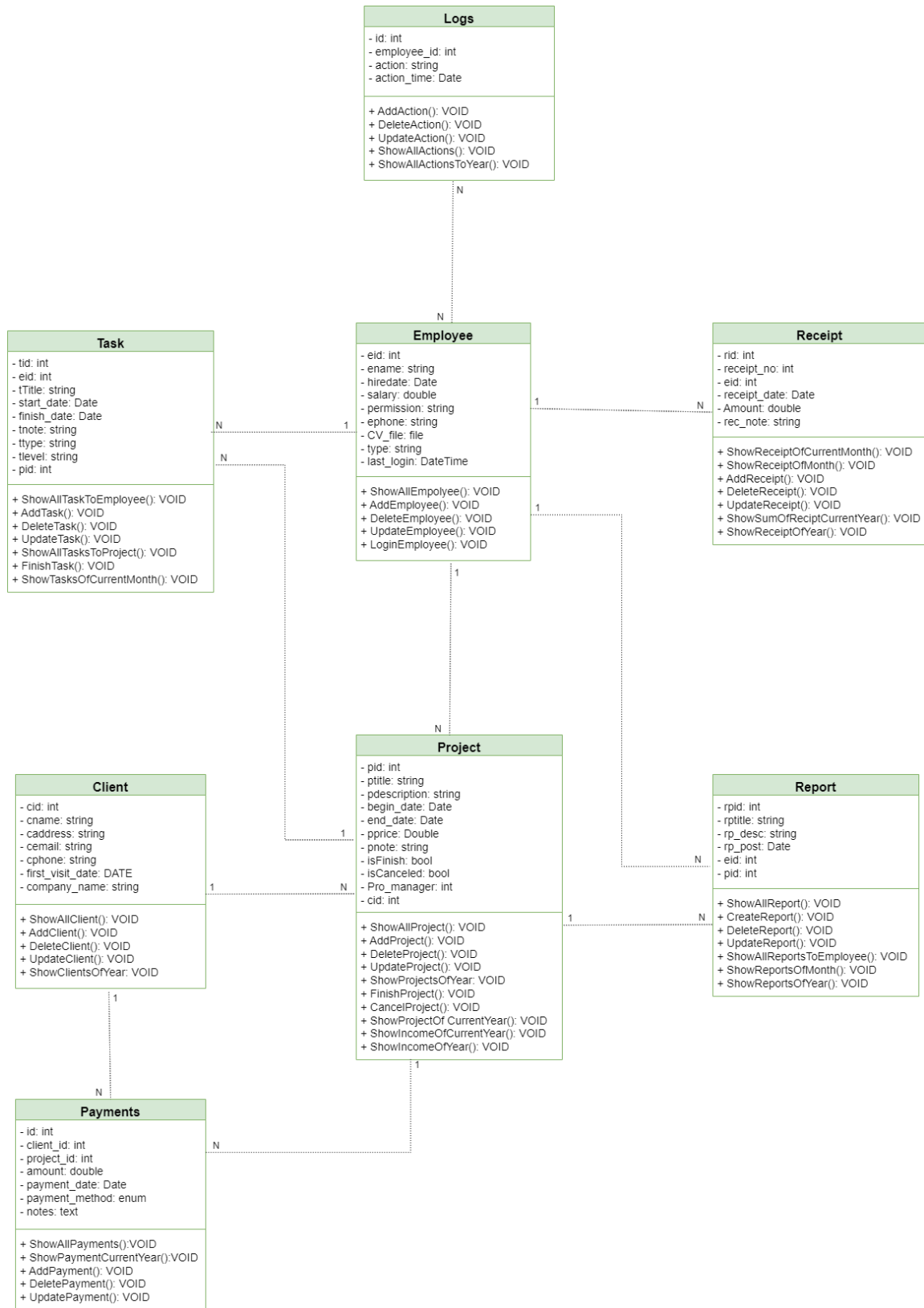
مخطط الكائنات العلائقي ERD:



مخطط حالة المستخدم UseCase:



مخطط الفئات :Class Diagram



Conceptual Scheme:

