University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: ___2____

Date: ____2025-03-02_____

Group Number: ____77_____

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Amy Xiong	97765291	m8o7v	amyxiongg@gmail.com
Matthew Haryanto	24695686	i7q9t	matthewanh10@gmail.com
Sadra Khosravi	90431511	o3v1b	sadrakh@outlook.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

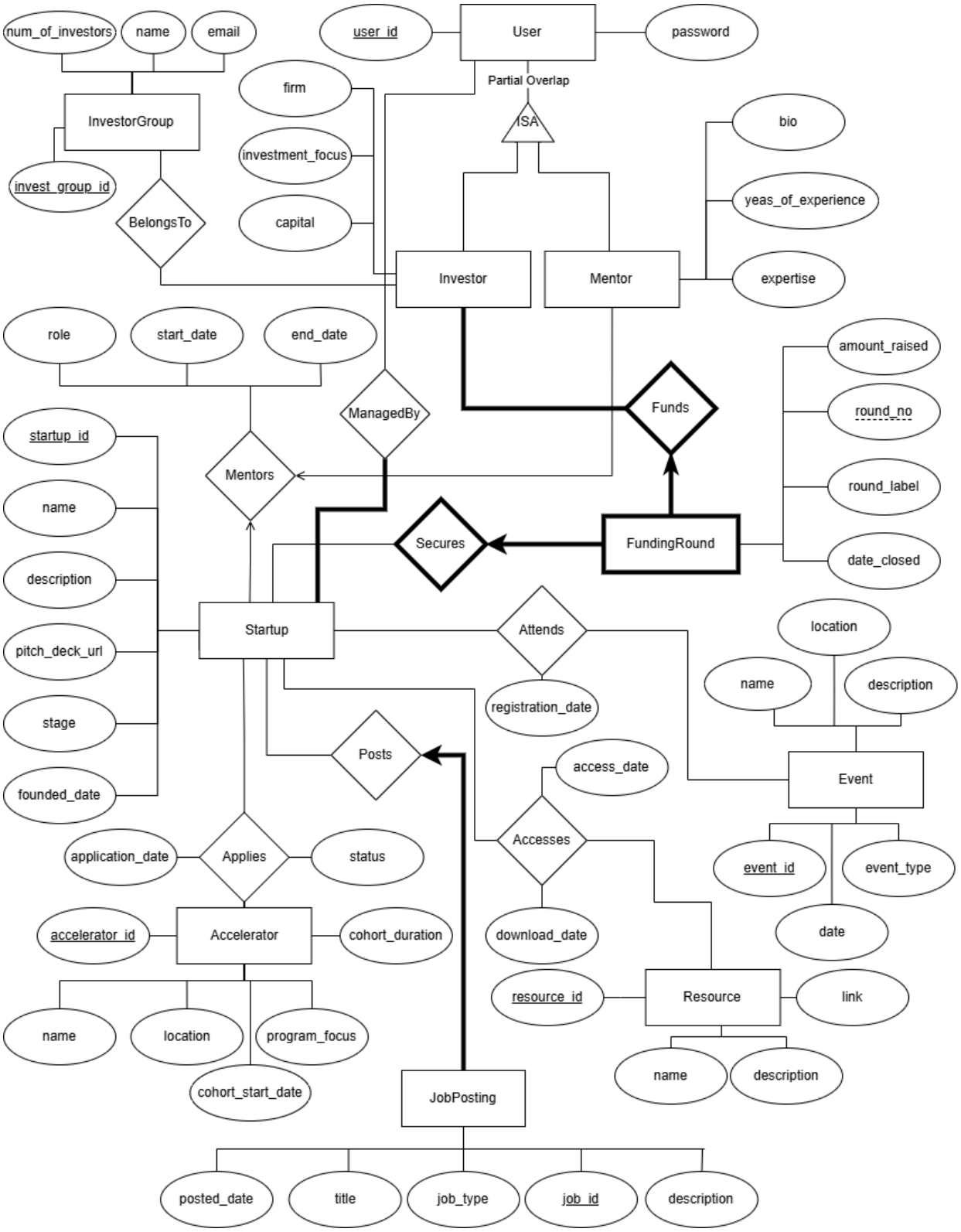
University of British Columbia, Vancouver
Department of Computer Science

Summary of Project (2-3 sentences):

Accelera is a platform that connects companies with investors, mentors, accelerators, and other vital resources in order to promote relationships within the startup ecosystem. The program simplifies important procedures including resource sharing, networking events, accelerator applications, mentoring access, and startup funding. The goal is to provide an effective way to handle the many interactions, encouraging growth of startups and collaboration among entrepreneurs.

University of British Columbia, Vancouver
Department of Computer Science

Database ER Diagram



Changes that were made:

- Add ManagedBy relationship, so that a user can manage a Startup instead of the Startup entity being its own entity
- Make FundingRound a weak entity to startup and investor, because the Investor would need to fund the FundingRound so it would be able to exist.
- Change the ISA constraint from total overlap to partial overlap, because a User can exist and manage a Startup without having to become an Investor or a Mentor

User Table

```
User(
    user_id: INT [PK],
    email: VARCHAR(100) [not null, unique],
    password: VARCHAR(100) [not null]
)
```

Candidate Keys (CK):

- user_id
- email

Primary Key (PK):

- user_id

Functional Dependencies:

- user_id → email, password
- email → user_id, password

Normalization:

- No need to further decompose, because it is already in BCNF.

University of British Columbia, Vancouver
Department of Computer Science

Investor Table (ISA relationship)

```
Investor(
    user_id: INT [PK, FK to User.user_id ON DELETE CASCADE],
    firm: VARCHAR(100),
    investment_focus: VARCHAR(50),
    capital: DECIMAL(15,2)
)
```

Candidate Keys (CK):

- user_id

Primary Keys (PK):

- user_id

Functional Dependencies:

- user_id → firm, investment_focus, capital

Normalization:

- Single attribute primary key.
- All other attributes depend on user_id.
- Therefore, Investor is in BCNF.

Mentor Table (ISA relationship and One-to-one Relationship with Startup)

```
Mentor(
    user_id: INT [PK, FK to User.user_id ON DELETE CASCADE],
    startup_id: INT [FK to Startup.startup_id, unique],
    bio: TEXT,
    years_of_experience: INT [not null],
    role: VARCHAR(100)
    start_date: DATE
    end_date: DATE
    expertise: VARCHAR(100) [not null]
)
```

Candidate Keys (CK):

- user_id

Primary Key (PK):

- user_id

Functional Dependencies:

- user_id → bio, years_of_experience, start_date, end_date, expertise, startup_id

Normalization:

- Single attribute primary key.
- All other attributes depend on user_id.
- Therefore, Investor is in BCNF.

Startup Table

```
Startup (
    startup_id: INT [PK],
    name: VARCHAR(100) [not null, unique],
    description: TEXT,
    pitch_deck_url: VARCHAR(100) [unique],
    stage ENUM('Idea', 'Seed', 'Series A', 'Series B', 'Growth',
'IPO') NOT NULL,
    founded_date: DATE [not null]
)
```

Candidate Keys (CK):

- name
- startup_id

Primary Key (PK):

- startup_id

Constraints:

- stage must only allow predefined ENUM values
- founded_date must be a valid date

University of British Columbia, Vancouver
Department of Computer Science

Functional Dependencies:

- startup_id → name, description, pitch_deck_url, stage, founded_date
- name → description, pitch_deck_url
- description → founded_date, stage
- pitch_deck_url → stage, founded_date

Steps:

1. Decompose on name → description, pitch_deck_url
2. Decompose on pitch_deck_url → stage, founded_date

Startup

- Attributes:
    - startup_id (PK)
    - Name

Startup is identified by a unique startup_id; the startup's name is also unique and will be used to relate to further attributes.

StartupAttributes

- Attributes:
    - name (PK)
    - description
    - pitch_deck_url (Unique)

Since name is a candidate key in the original table, it uniquely determines the description and the pitch deck URL. (Note that we assume each startup's name is unique.)

StartupDetails

- Attributes:
    - pitch_deck_url (PK)
    - stage
    - founded_date

With pitch_deck_url being unique, it can determine the stage and founding date. By using pitch_deck_url as the key for this relation, we isolate this dependency.

With this decomposition we are both in BCNF and 3NF.

University of British Columbia, Vancouver
Department of Computer Science

Accelerator Table

```
Accelerator (
    accelerator_id: INT [PK],
    name: VARCHAR(500) [not null],
    location: VARCHAR(500) [not null],
    cohort_start_date: DATE [not null],
    program_focus: TEXT,
    unique (name, location, cohort_start_date)
)
```

Candidate Keys (CK):

- accelerator_id
- name, location, cohort_start_date

Primary Key (PK):

- accelerator_id

Constraints:

- cohort_start_date must be a valid date
- name should be unique across accelerators

Functional Dependencies:

- accelerator_id → name, location, cohort_start_date, program_focus
- name, location, cohort_start_date → accelerator_id, program focus

Normalization:

These FDs ensure that every non-key attribute is fully functionally dependent on a candidate key, keeping the Accelerator table in BCNF (and therefore in 3NF).

Applies Table

```
Accelerator (
    accelerator_id: INT [PK, FK to Accelerator.accelerator_id],
    startup_id: INT [PK, FK to Startup.startup_id],
    application_date: DATE,
```

```
    status: VARCHAR(10)
)
```

Candidate Keys (CK):

- (accelerator_id, startup_id)

Primary Key (PK):

- (accelerator_id, startup_id)

Functional Dependencies:

- accelerator_id → name, location, cohort_start_date, program_focus

Normalization:

- No need to further decompose, because it is already in BCNF.

BelongsTo Table (Many-to-many relationship)

```
BelongsTo(
    user_id: INT [PK, FK to Investor.user_id],
    invest_group_id: INT [PK, FK to InvestorGroup.invest_group_id],
)
```

Candidate Keys (CK):

- (user_id, invest_group_id)

Primary Key (PK):

- (user_id, invest_group_id)

Functional Dependencies:

- (user_id, invest_group_id) → (user_id, invest_group_id)

Normalization:

- No need to further decompose, because it is already in BCNF.

University of British Columbia, Vancouver
Department of Computer Science

ManagedBy Table (Many-to-Many Relationship)

```
ManagedBy(
    user_id: INT [PK, FK to User.user_id ON DELETE RESTRICT],
    startup_id: INT [PK, FK to Startup.startup_id ON DELETE CASCADE],
    role: VARCHAR(50) [NOT NULL],
    start_date: DATE [NOT NULL],
    CONSTRAINT valid_role CHECK (role IN ('Founder', 'Co-Founder',
'CEO', 'CTO', 'Employee', 'Advisor'))
)
```

Candidate Keys (CK):

- (user_id, startup_id)

Primary Key (PK):

- (user_id, startup_id) - Composite key

Functional Dependencies:

- (user_id, startup_id) → role, start_date
- (startup_id, role) → user_id (Non-PK/CK FD: certain roles like 'CEO' can only be held by one user per startup)

Normalization:

- No need to further decompose, because it is already in BCNF.

Additional Constraints:

- Every startup_id in Startup must appear at least once in the ManagedBy table (enforcing total participation)
- This will need to be implemented through either:
    - Database triggers
    - Application-level logic
    - CHECK constraints with subqueries (if supported by the DBMS)

FundingRound Table (Weak Entity)

```
SecuresFundingRound(
    round_no: INT,
    amount_raised: INT,
    round_label: VARCHAR(100),
    date_closed: DATE,
    user_id: INT [FK to Investor.user_id],
    startup_id: INT [FK to Startup.startup_id],
    PK(round_no, user_id, startup_id)
)
```

Candidate Keys (CK):

- (startup_id, user_id, round_no)

Primary Key (PK):

- (startup_id, user_id, round_no)

Functional Dependencies:

- funding_round_id → startup_id, round_type, amount, funding_date

Normalization:

- No need to further decompose, because it is already in BCNF.

JobPostingPosts Table (Many-to-One Relationship)

```
JobPostingPosts(
    job_id: INT [PK],
    title: VARCHAR(10),
    description: TEXT,
    posted_date: DATE,
    job_type: VARCHAR(10),
    startup_id: INT [FK to Startup.startup_id, unique]
)
```

Candidate Keys (CK):

- (job_id)

Primary Key (PK):

- (job_id)

Functional Dependencies:

- (job_id) → (startup_id, title, description, posted_date)

Normalization:

- No need to further decompose, because it is already in BCNF.

Accesses Table (Many-to-Many Relationship)

```
Accesses(
    access_date: INT,
    download_date: VARCHAR(10),
    resource_id: INT [PK, FK to Resource.resouce_id],
    startup_id: INT [PK, FK to Startup.startup_id],
)
```

Candidate Keys (CK):

- (resource_id, startup_id)

Primary Key (PK):

- (resource_id, startup_id)

Functional Dependencies:

- (resource_id, startup_id) → (access_date, download_date)

Normalization:

- No need to further decompose, because it is already in BCNF.

Resource Table

```
Resource(
    resource_id: INT [PK],
```

```
    name: VARCHAR(500),
    description: VARCHAR(500),
    link: VARCHAR(100)
)
```

Candidate Keys (CK):

- (resource_id)

Primary Key (PK):

- (resource_id)

Functional Dependencies:

- (resource_id) → (name, description, link)

Normalization:

- No need to further decompose, because it is already in BCNF.

Attends Table (Many-to-Many Relationship)

```
Attends(
    event_id: INT [PK, FK to Event.event_id],
    startup_id: INT [PK, FK to Startup.startup_id],
    registration_date: DATE,
)
```

Candidate Keys (CK):

- (event_id, startup_id)

Primary Key (PK):

- (event_id, startup_id)

Functional Dependencies:

- (event_id, startup_id) → (registration_date)

No need to further decompose, because it is already in BCNF.

University of British Columbia, Vancouver
Department of Computer Science

Event Table

```
Event(
    event_id: INT [PK],
    name: VARCHAR(100),
    location: VARCHAR(100),
    description: VARCHAR(500),
    event_type: VARCHAR(30)
)
```

Candidate Keys (CK):

- (event_id)

Primary Key (PK):

- (event_id)

Functional Dependencies:

- (event_id) → (name, location, description, event_type)

No need to further decompose, because it is already in BCNF.

SQL DDL

```
CREATE TABLE Accelerator(
     accelerator_id      INT PRIMARY KEY,
     name                VARCHAR(500) NOT NULL,
     location            VARCHAR(500) NOT NULL,
     cohort_start_date   DATE NOT NULL,
     program_focus       TEXT,
     UNIQUE(name, location, cohort_start_date)
)

CREATE TABLE Applies(
     accelerator_id      INT,
     startup_id          INT
     application_date    DATE,
     status              VARCHAR(10)
```

```
      PRIMARY KEY(accelerator_id, startup_id)
      FOREIGN KEY(accelerator_id)
                REFERENCES Accelerator(accelerator_id)
      FOREIGN KEY (startup_id)
                REFERENCES Startup(startup_id)
)

CREATE TABLE User(
      user_id     INT PRIMARY KEY,
      email       VARCHAR(100) NOT NULL UNIQUE,
      password    VARCHAR(100) NOT NULL
)
CREATE TABLE Investor(
      user_id           INT,
      firm              VARCHAR(100),
      invesment_focus   VARCHAR(50),
      capital           DECIMAL(15, 2),
      PRIMARY KEY(user_id),
      FOREIGN KEY(user_id)
                REFERENCES User(user_id)
                ON DELETE CASCADE
)
CREATE TABLE Mentor(
      user_id             INT PRIMARY KEY,
      startup_id          INT UNIQUE,
      bio                 TEXT,
      years_of_experience INT NOT NULL,
      role                VARCHAR(100),
      start_date          DATE,
      end_date            DATE,
      expertise           VARCHAR(100) NOT NULL,
      FOREIGN KEY(user_id)
                REFERENCES User(user_id)
                ON DELETE CASCADE,
      FOREIGN KEY(startup_id)
                REFERENCES Startup(startup_id)
)

CREATE TABLE Startup(
      startup_id      INT PRIMARY KEY
      name             VARCHAR(100)
      FOREIGN KEY(name)
                REFERENCES StartupAttributes(name)
)
```

```
CREATE TABLE StartupAttributes(
     name           VARCHAR(100) PRIMARY KEY,
     description     TEXT,
     pitch_deck_url  VARCHAR(100) UNIQUE
     FOREIGN KEY(pitch_deck_url)
               REFERENCES StartupDetails(pitch_deck_url)
)

CREATE TABLE StartupDetails(
     pitch_deck_url   VARCHAR(100) PRIMARY KEY,
     stage            VARCHAR(100),
     founded_date     DATE
)

CREATE TABLE BelongsTo(
     user_id          INT,
     invest_group_id  INT,
     PRIMARY KEY(user_id, invest_group_id),
     FOREIGN KEY(user_id)
               REFERENCES Investor(user_id)
     FOREIGN KEY(invest_group_id)
               REFERENCES InvestorGroup(invest_group_id)
)

CREATE TABLE ManagedBy(
     user_id     INT NOT NULL,
     startup_id  INT NOT NULL,
     role        VARCHAR(50) NOT NULL
                     CHECK (role IN ('Founder', 'Co-Founder', 'CEO',
                     'CTO', 'Employee', 'Advisor')),
     start_date  DATE NOT NULL,
     PRIMARY KEY(user_id, startup_id),
     FOREIGN KEY(user_id)
               REFERENCES User(user_id),
     FOREIGN KEY (startup_id)
               REFERENCES Startup(startup_id)
)

CREATE TABLE FundingRound(
     round_no       INT,
     amount_raised  INT,
     round_label    VARCHAR(100),
     date_closed    DATE,
```

```
      user_id        INT,
      startup_id     INT,
      PRIMARY KEY(round_no, user_id, startup_id),
      FOREIGN KEY(user_id)
                REFERENCES Investor(user_id)
                ON DELETE CASCADE
                ON UPDATE CASCADE,
      FOREIGN KEY(startup_id)
                REFERENCES Startup(startup_id)
                ON DELETE CASCADE
                ON UPDATE CASCADE
)

CREATE TABLE JobPostingPosts(
      job_id         INT PRIMARY KEY,
      title          VARCHAR(10),
      description    TEXT,
      posted_date    DATE,
      job_type       VARCHAR(10),
      startup_id     INT UNIQUE,
      FOREIGN KEY(startup_id)
                REFERENCES Startup(startup_id)
)

CREATE TABLE Accesses(
      access_date    INT,
      download_date  VARCHAR(10),
      resource_id    INT,
      startup_id     INT,
      PRIMARY KEY(resource_id, startup_id),
      FOREIGN KEY(startup_id)
                REFERENCES Startup(startup_id),
      FOREIGN KEY(resource_id)
                REFERENCES Resource(resource_id)
)

CREATE TABLE Resource(
      resource_id    INT PRIMARY KEY,
      name           VARCHAR(500),
      description    TEXT,
      link           VARCHAR(100)
)

CREATE TABLE Attends(
```

```
      event_id           INT,
      startup_id         INT,
      registration_date  DATE,
      PRIMARY KEY(event_id, startup_id),
      FOREIGN KEY(event_id)
                REFERENCES Event(event_id),
      FOREIGN KEY(startup_id)
                REFERENCES Startup(startup_id)
)

CREATE TABLE Event(
      event_id       INT PRIMARY KEY,
      name           VARCHAR(100),
      location       VARCHAR(100),
      description    VARCHAR(500),
      event_type     VARCHAR(30)
)
```

Populating Tables

```
INSERT
INTO Accelerator(accelerator_id, name, location, cohort_start_date,
program_focus)
VALUES
      (1, 'LaunchPad', 'Toronto', '2024-04-01', 'AI & Robotics'),
      (2, 'Capital Boost', 'Ottawa', '2024-05-15', 'Blockchain'),
      (3, 'Startup Works', 'Vancouver', '2024-06-10', 'E-commerce &
      Retail'),
      (4, 'Green Future', 'Seattle', '2024-07-20', 'Sustainable
      Tech'),
      (5, 'Health Forward', 'Richmond', '2024-08-05', 'Health &
      Wellness');

(before BCNF decomposition)
INSERT INTO Startup (startup_id, name, description, pitch_deck_url,
stage, founded_date)
VALUES
      (1, 'MealMate', 'Meal planning app with grocery integration',
      'mealmate_pitch.pdf', 'Seed', '2023-02-14'),
      (2, 'SmartWatts', 'Home energy usage tracker and optimizer',
      'smartwatts_pitch.pdf', 'Series A', '2021-08-30'),
```

```
     (3, 'WellBand', 'Smartwatch that tracks senior health metrics',
     'wellband_pitch.pdf', 'Series B', '2020-06-25'),
     (4, 'PlayCloud', 'Cloud gaming platform for indie developers',
     'playcloud_pitch.pdf', 'Growth', '2019-12-10'),
     (5, 'QuickDrop', 'Drone delivery service for online orders',
     'quickdrop_pitch.pdf', 'IPO', '2018-04-05');

(after BCNF decomposition)
INSERT INTO Startup (startup_id, name)
VALUES
(1, 'MealMate'),
(2, 'SmartWatts'),
(3, 'WellBand'),
(4, 'PlayCloud'),
(5, 'QuickDrop');


INSERT
INTO StartupAttributes (name, description, pitch_deck_url)
VALUES
     ('MealMate', 'Meal planning app with grocery integration',
     'mealmate_pitch.pdf'),
     ('SmartWatts', 'Home energy usage tracker and optimizer',
     'smartwatts_pitch.pdf'),
     ('WellBand', 'Smartwatch that tracks senior health metrics',
     'wellband_pitch.pdf'),
     ('PlayCloud', 'Cloud gaming platform for indie developers',
     'playcloud_pitch.pdf'),
     ('QuickDrop', 'Drone delivery service for online orders',
     'quickdrop_pitch.pdf');


INSERT
INTO StartupDetails (pitch_deck_url, stage, founded_date)
VALUES
('mealmate_pitch.pdf', 'Seed', '2023-02-14'),
('smartwatts_pitch.pdf', 'Series A', '2021-08-30'),
('wellband_pitch.pdf', 'Series B', '2020-06-25'),
('playcloud_pitch.pdf', 'Growth', '2019-12-10'),
('quickdrop_pitch.pdf', 'IPO', '2018-04-05');



INSERT
```

```
INTO Applies (accelerator_id, startup_id, application_date, status)
VALUES
    (3, 1, '2024-02-01', 'Pending'),
    (1, 2, '2024-01-10', 'Rejected'),
    (2, 3, '2024-01-22', 'Accepted'),
    (5, 4, '2024-02-15', 'Accepted'),
    (4, 5, '2024-03-02', 'Pending');


INSERT
INTO User (user_id, email, password)
VALUES
    (1, 'lisa@gmail.com', 'strongP@ssw0rd'),
    (2, 'mark@gmail.com', 'random123'),
    (3, 'nina@gmail.com', 'health4all!'),
    (4, 'jake@gmail.com', 'gamingMaster42'),
    (5, 'sophia@gmail.com', 'drones2025');


INSERT
INTO Investor (user_id, firm, invesment_focus, capital)
VALUES
    (1, 'SkyBridge Capital', 'AI & Robotics', 8000000.00),
    (2, 'Blue Ocean Ventures', 'Green Energy', 3200000.50),
    (3, 'Quantum Capital', 'FinTech', 12000000.75),
    (4, 'Summit Growth Fund', 'E-commerce & Retail', 5500000.00),
    (5, 'Pioneer VC', 'SaaS & Cloud Computing', 20000000.00);


INSERT
INTO Mentor (user_id, startup_id, bio, years_of_experience, role,
start_date, end_date, expertise)
VALUES
     (1, 1, 'Former AI researcher at Google, now mentoring startups
     in deep learning applications.', 15, 'Technical Advisor',
     '2023-05-01', NULL, 'Machine Learning'),
     (2, 2, 'Ex-CTO of a green tech startup, specializing in
     sustainability and energy optimization.', 10, 'Industry
     Mentor', '2022-08-15', NULL, 'Sustainable Technology'),
     (3, 3, 'Angel investor turned mentor, guiding fintech startups
     on scaling and compliance.', 12, 'Investor Mentor',
     '2023-01-10', NULL, 'FinTech'),
```

```
    (4, 4, 'Former VP of Marketing at a Fortune 500, helping
    startups with growth strategies.', 18, 'Marketing Consultant',
    '2021-11-20', '2024-02-28', 'Growth & Marketing'),
    (5, 5, 'Blockchain developer with extensive experience in
    decentralized finance.', 8, 'Blockchain Advisor', '2022-03-05',
    NULL, 'Blockchain & Crypto');


INSERT
INTO BelongsTo (user_id, invest_group_id)
VALUES
    (1, 201),
    (3, 202),
    (4, 203),
    (2, 201),
    (5, 204);


INSERT
INTO ManagedBy (user_id, startup_id, role, start_date)
VALUES
    (1, 1, 'Founder', '2023-02-14'),
    (2, 2, 'CEO', '2021-09-01'),
    (3, 3, 'Co-Founder', '2020-07-01'),
    (4, 4, 'CTO', '2019-12-15'),
    (5, 5, 'CEO', '2018-05-01');


INSERT
INTO FundingRound (round_no, amount_raised, round_label, date_closed,
user_id, startup_id)
VALUES
    (1, 150000, 'Pre-Seed', '2023-03-22', 1, 1),
    (2, 3000000, 'Series A', '2022-09-05', 2, 2),
    (3, 12000000, 'Series B', '2021-11-20', 3, 3),
    (4, 28000000, 'Growth', '2020-08-30', 4, 4),
    (5, 75000000, 'IPO', '2019-06-15', 5, 5);


INSERT
INTO JobPostingPosts (job_id, title, description, posted_date,
job_type, startup_id)
VALUES
```

```
      (1, 'ML Engineer', 'Build AI models for meal planning',
      '2024-03-01', 'Full-Time', 1),
      (2, 'Frontend Dev', 'Develop web dashboard for energy
      monitoring', '2024-02-15', 'Part-Time', 2),
      (3, 'UX Designer', 'Improve accessibility for wearable
      devices', '2024-02-25', 'Full-Time', 3),
      (4, 'Game Dev', 'Work on multiplayer cloud games',
      '2024-03-05', 'Contract', 4),
      (5, 'Logistics Lead', 'Optimize drone delivery routes',
      '2024-03-10', 'Full-Time', 5);


INSERT
INTO Accesses (access_date, download_date, resource_id, startup_id)
VALUES
    (2024-02-05, '2024-02-05', 101, 1),
    (2024-02-08, '2024-02-08', 102, 2),
    (2024-02-10, '2024-02-10', 103, 3),
    (2024-02-12, '2024-02-12', 104, 4),
    (2024-02-15, '2024-02-15', 105, 5);


INSERT
INTO Resource (resource_id, name, description, link)
VALUES
      (101, 'Lean Startup Ebook', 'Guide to running a lean startup',
      'lean_startup.pdf'),
      (102, 'Blockchain Whitepaper', 'Introduction to decentralized
      apps', 'blockchain_whitepaper.pdf'),
      (103, 'Investor Deck Template', 'Template for pitching to VCs',
      'investor_deck.pptx'),
      (104, 'SEO Playbook', 'Strategies for online growth',
      'seo_guide.pdf'),
      (105, 'Legal Handbook', 'Common legal pitfalls for startups',
      'startup_legal.pdf');


INSERT
INTO Attends (event_id, startup_id, registration_date)
VALUES
    (101, 1, '2024-04-01'),
    (102, 2, '2024-04-02'),
    (103, 3, '2024-04-05'),
    (104, 4, '2024-04-10'),
```

```
    (105, 5, '2024-04-12');


INSERT
INTO Event (event_id, name, location, description, event_type)
VALUES
     (101, 'Networking', 'Surrey', 'Meet early-stage startup
     founders', 'Meetup'),
     (102, 'Crypto', 'Richmond', 'Panel on blockchain trends',
     'Conference'),
     (103, 'Retail Innovation', 'Burnaby', 'Showcase of new retail
     tech', 'Expo'),
     (104, 'Tech Tips', 'Coquitlam', 'Talks on sustainable
     startups', 'Conference'),
     (105, 'Funding', 'Vancouver', 'Investors scouting for
     startups', 'Networking');
```

AI Declaration

No AI tools were used in this assignment.