Final Fantasy All-Stars: Project Step 3 Final

CS 340 - Intro to Databases
Team Debuggers for Donuts

URL: http://flip2.engr.oregonstate.edu:50000/index.html

Sara Biebuyck
Patrick Cheng

a) Feedback

Step 3 Draft:

Review 1

Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.

No, the intersection tables are not shown in the UI. I know it's still early, so you might already be working on it, but having a more detailed character view would be nice. You could show more attributes like their abilities and other nice to have info in one spot.

Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?

Yes, more than one of the SELECT queries utilize a filter.

Does the UI implement an INSERT for every table in the schema? In other words, there should be UI input fields that correspond to each table and attribute in that table.

Yes, every UI page has an INSERT, though some tables are not represented in the UI. Most of the actions pages are not present.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? In other words if there is a M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price and line_total).

To be honest, this is a hard one to really tell. The BattleActions looks like it is set up properly. Most of the intersection tables are not on the UI at this time.

Is there at least one DELETE and does at least one DELETE remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.

Yes, there is at least one DELETE. Again, no intersection tables are visible on the UI at this time.

Is there at least one UPDATE for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single

ProductID record?

Yes, there are many UPDATEs shown in the UI.

Is at least one relationship NULLable? In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.

Yes, Characters and Spells have an optional relationship

Do you have any other suggestions for the team to help with their HTML UI? For example using AS aliases to replace obscure column names such as fname with First Name.

I feel like this is a hard assignment and has some vague requirements with material that I don't fully grasp. But you guys did a great job. I am excited to see your final project.

Add some styling to the site, format the nav bar. look at bootstrap if you haven't already

add the intersection tables to the project

think about renaming some of the names in the navbar for better readability.

Review 2

· *Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.*

The UI does not use SELECT for every table in the schema. The intersection tables Ui for the outline does not show such as SpellActions, Ability Actions, and CharacterActions.

· *Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?*

Yes, there is at least one SELECT that utilizes a search with a dynamically populated list of properties in DDL.sql. However, the DML.sql do not include in the zip file. You should add it to the further assignment.

· *Does the UI implement an INSERT for every table in the schema? In other words, there should be UI input fields that correspond to each table and attribute in that table.*

Yes, the UI does implement an INSERT for every table in the schema.

· Does each INSERT also add the corresponding FK attributes, including at least one M: M relationship? In other words, if there is an M: M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price, and line_total).

No, not really. Intersection tables in the outline are all not in the UI. You should create them in the UI and add the corresponding FK attributes, including at least one M: M relationship.

· Is there at least one DELETE and does at least one DELETE remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.

Yes every table created in the UI has a DELETE. However, it has to say that there is at least one DELETE removes things from an M:M relationship since the intersection table used to join M: M relationships are not in the UI.

· Is there at least one UPDATE for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?

Yes, every table created in the UI has an UPDATE. For example, the user can update characterID in the BattleActions.

· Is at least one relationship NULLable? In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.

Yes, there is at least one relationship NULLable. For example, in the Battle Actions table page, spellID has some NULL values.

· Do you hve any other suggestions for the team to help with their HTML UI? For example using AS aliases to replace obscure column names such as fname with First Name.

1. Arrange the table vertically on the home page so that it looks more convenient.

2. Add DML.sql in your zip file.

3. Home page and Homes table page are too similar. You can change the name of the Home page to the Main page.

4. Add all intersection tables in the UI.

Overall, nice work.

Review 3

Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.

All the entity tables are represented, but I don't see how the intersection tables are present. For example, there are no Abilities represented on the Character and no Characters on the Abilities page, meaning AbilityActions is not represented.

Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?

Yes, several of the SELECT queries utilize a filter. For example:

```
INSERT INTO CharacterActions(characterID, actionID)
VALUES
((SELECT characterID FROM Characters WHERE characterName = 'Lavos'), 1),
((SELECT characterID FROM Characters WHERE characterName = 'Cloud Strife'), 2),
((SELECT characterID FROM Characters WHERE characterName = 'Terra'), 3);
```

Does the UI implement an INSERT for every table in the schema? In other words, there should be UI input fields that correspond to each table and attribute in that table.

For all the tables that are represented, there are INSERT functions present in the UI.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? In other words if there is a M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price and line_total).

It is hard to tell - the INSERT functions for the M:M relationships show some FK, for example spellID, abilityID, characterID on BattleActions, but not all intersection tables are represented.

Is there at least one DELETE and does at least one DELETE remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.

There are many DELETEs. Similarly to above it is not completely clear of how the

DELETE will behave from the UI, but it looks correct from the code.

Is there at least one UPDATE for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?

Yes, there are many UPDATEs shown in the UI. I don't see code corresponding to this, so I'm unsure if the changes would be handled correctly.

Is at least one relationship NULLable? In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.

Yes - we can see evidence of this in BattleActions page.

Do you have any other suggestions for the team to help with their HTML UI? For example using AS aliases to replace obscure column names such as fname with First Name.

I would change the "Home" button to something else, like Index, or put it on a different row from the table selection. I clicked on Home a few times thinking it was the table for Homes.

<u>Review 4</u>

Does the UI utilize a SELECT for every table in the schema? In other words, data from each table in the schema should be displayed on the UI. Note: it is generally not acceptable for just a single query to join all tables and displays them.

The UI didn't utilize every table in the schema. It would help if you utilized the SpellActions, AbilityActions, and CharacterActions. You can add other pages to access those tables.

Does at least one SELECT utilize a search/filter with a dynamically populated list of properties?

At least one is selected to utilize a search/filter with a dynamically populated list of properties. In the battle action table, you use search/filter to display the data.

Does the UI implement an INSERT for every table in the schema? In other words, there should be UI input fields that correspond to each table and attribute in that table.

UI implements an Insert for every table on UI, but the insert is missing out for SpellActions, AbilityActions, and CharacterActions. Please add a text box and components. The insert already has preset information.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? In other words if there is a M:M relationship between Orders and Products, INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price and line_total).

The Battle Actions has an insert that adds to the corresponding FK attributes. You can add the character, spell, ability, and action to the table.

Is there at least one DELETE and does at least one DELETE remove things from a M:M relationship? In other words, if an order is deleted from the Orders table, it should also delete the corresponding rows from the OrderDetails table, BUT it should not delete any Products or Customers.

There are at least Delete and at least Deletes to remove the thing from M: M relationship. The SQL code also contains a cascade on delete.

Is there at least one UPDATE for any one entity? In other words, in the case of Products, can productName, listPrice, qtyOnHand, e.g. be updated for a single ProductID record?

There's at least an update for any entity. We can update the attributes with the ID. However, all the update is present and can't be changed.

Is at least one relationship NULLable? In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus it should be feasible to edit an Order and change the value of Employee to be empty.

There are at least relationships NULLable. The relationship between spellID and BattleAction can be NULL.

and change the value of Employee to be empty.

Do you have any other suggestions for the team to help with their HTML UI? For example using AS aliases to replace obscure column names such as fname with First Name.

Recommendation on the design:

You can add an input text and markdown box to the selection.

The naming is confusing on the UI. Example: Homes and Home.

It would be best if you had more time to work out CSS detail properly.

TA Review
I would prefer not to merge the relationship lines from/to an entity in both ERD and Schema.

Couldn't find the intersection tables on the pages. Either they should have different pages or should be included on one of the related tables' pages.

Actions to Take/Taken
The intersection tables were not present in the database design at all. They need to be created from the ground up, included in the outline/diagrams, sample data created, then included in the website.

Styling will be included eventually. Content is still evolving too quickly to focus on styling because one depends on the other. Eventually:

- Home page -> Main page for clarity to avoid conflict with Homes page
- Different styling to set things apart
- Arrange tables vertically for easier reading
- CSS

DMLs were omitted initially but were included eventually.

Unoverlap the ERD/schema relationship lines.

<u>Step 2 Draft:</u>

Review 1
Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

No, in their ERD it shows having three M:N relationships but there are no intersection tables shown in their schema. If the intent is to have BattleActions have only M:N relationships, then intersection tables need to be added for the relationships with Characters, Spells, and Abilities.

Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

One small typo: varchat in the ER diagram

For the ER diagram, BattleActions to Characters/Spells/Abilities has a M:N relationship but contains multiple foreign keys. To fix this, I’d either make them a M:1 relationship or remove the foreign keys (and use intersection tables).

Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

The schema is easy to read and there are no lines crossing.

Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

There are currently no intersection tables between the M:N relationships outlined in the ER diagram.

Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

No the sample data doesn’t suggest any non-normalized issues.

Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

The SQL file is syntactically correct; the sample data for the Characters table requires a non NULL inParty and inBattle Boolean and there is no default, but the insert doesn’t include values for it so it seems to be failing.

In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

Almost all of the data types in the SQL are appropriate based on the description in the outline except for actionEffect in the BattleActions table (it’s marked as text in SQL but should be a Boolean based on the database outline).

In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

The primary and foreign keys are correct compared to the schema. There are no CASCADE operations but there are no intersection tables where they would apply.

In the SQL, are relationship tables present when compared to the ERD/Schema?

There are no relationship tables present, but there are also no intersection tables defined in the schema.

In the SQL, is all example data shown in the PDF INSERTED?

No, some data is not inserted due to the issue mentioned above for Characters.

Review 2
Great job guys!! Such a cool idea!! I enjoyed reading through your outline!!

¨ Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

Yes, the schema presents an accurate model that follows the database outline and ER logical diagram.

¨ Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

a) Yes, there is consistency in naming, however there is typo with varchar showing as 'varchat'… Probably autocorrect issue.

b) Yes, there is consistency between entities (plural) and attributes (singular).

c) Yes, there is consistency in capitalization.

¨ Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

Yes, the schema is very easy to read.

¨ Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

I did not see any intersection tables. I do see Spells and Characters as having a M:M relationship. Also, Battleships has a M:M with Spell, Actions, and Characters. These are possible intersection tables.

¨ Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

In the sample data I see multiple 'spellTypes' that are the same listed under Spells that could be a possible issue.

¨ Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

Yes, the SQL file is syntactically correct.

¨ In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

I didn’t see any issues.

¨ In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

Yes, the primary and foreign keys are correctly defined when compared to the Schema. I did not see any cascade operations declared.

¨ In the SQL, are relationship tables present when compared to the ERD/Schema?

Yes, the relationship tables are present in the SQL.

¨ In the SQL, is all example data shown in the PDF INSERTED?

Yes, example data is shown.

Review 3
Ashley Calton

This is a great idea, I really enjoyed looking through your project.

Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

Yes, the schema presents a physical model that follows the database outline and ERD exactly.

Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

a) Yes, the naming is consistent between each section.

b) Yes, the entities are plural and the attributes are singular.

c) Yes, the entities are in PascalCase and the attributes in camelCase.

Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

Yes, the schema is easy to read and there are no lines crossed.

Small typo - all varchar are spelled 'varchat'

Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

None of the tables are referred to as an intersection table, but BattleActions seems to fulfill the intersection table role. BattleActions has three FKs and faciliates the multiple M:N relationships, as it represents a single BattleAction based on one Character, Spell, and Ability.

Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

I see no issues.

Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

Yes, the SQL file is syntactically correct.

I would add the disable commits and foreign key checks suggested on the assignment page, I have found it to be helpful when rerunning the file

In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

BattleActions: actionEffect is describe as a bool in the outline, but it is text describing the effect in the SQL.

In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

The primary and foreign keys are correct compared to the schema.

There are no CASCADE operations declared.

In the SQL, are relationship tables present when compared to the ERD/Schema?

Yes, the relationship tables are present compared to the ERD/schema.

In the SQL, is all example data shown in the PDF INSERTED?

Yes, the example data in the PDF is identical to that of the SQL.

Review 4
Michael Ver Voort

Hi Group 41! Great idea - definitely fun and different. I have some great childhood memories of classic JRPGs and really enjoyed reading about your project.

Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

No, some relationships that are in the outline and ERD appear to be missing from the schema. It's not clear to me from the schema how those relationships from the ERD/outline are implemented. For example:

The outline and ERD indicate a M:M relationship between Spells and Characters and between Spells and BattleActions, but the schema shows what appears to be a 1:M relationship with BattleActions (via FK in BattleActions).

The ERD shows a 1:1 relationship between Abilities and Characters but there is no link between them in the schema.

Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Yes. All of the naming is consistent as far as I can see.

Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

Yes, it's easy to see what's going on.

Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

In the schema, BattleActions is an intersection table that facilitates a M:N relationship between each of Characters, Spells, and Abilities. There are a few other M:N relationships indicated in the ERD that do not have corresponding intersection tables in the schema.

Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

None that I can find.

Is the SQL file syntactically correct? This can be easily verified by using PhPMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

Yep - everything runs smoothly.

In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

In the table BattleActions, "actionEffect" is a text field in the SQL but a boolean value everywhere in the PDF.

In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

The keys are all correct, but there are no CASCADE operations declared.

In the SQL, are relationship tables present when compared to the ERD/Schema?

As mentioned, the ERD and schema conflict; the SQL file matches the schema.

In the SQL, is all example data shown in the PDF INSERTED?

Yes, all of the sample data is the same.

Other thoughts: after reading some of the corrections from the previous draft, I think the schema is closer to what you are intending to implement, and you just need to correct the ERD and outline to match the schema.

TA Review

The way I think is better and clear is if you have verbatim feedbacks from peers and TA in one subsection, then the actions you took in regard to those feedbacks in the next one and finally the upgrades that you made in this step (updates you made on your own) just like in the instructions.

Step 1 draft:
Peer feedbacks:
Review 1
Review 2
Review 3
Actions taken
Requests, Changes and justification if not taken
Upgrades

Actions to take/taken

Biggest changes are that we need to add many intersection tables in different places. Currently not displayed. Also need to add the 1:1 Abilities/Characters relationship in the schema. Need FKs to link them. Will add CASCADE operations to appropriate SQL statements.

Need to correct the “varchat” typo.

SpellTypes is not currently causing any dependencies. In the future for added functionality, it is possible to create another Category Table to extend labeling for Magic Types to Abilities or Characters, or some other unforeseen entity, but for now this is beyond our scope and deemed mission creep.
Need to change inParty and inBattle null settings, and add lines for commits/key checks per comments.

Will update outline to reflect text setting of BattleActions effect attribute.

<u>Step 1 Draft:</u>

1. *Does the overview describe what problem is to be solved by a website with DB back end?*

AB: A problem isn’t specifically defined, but the functionality for the database is described. Perhaps the last sentence can be expanded to further emphasize the database’s ability to streamline access to specific information for Square Enix to better address their problem of sifting through so many characters and games, for example.

RESPONSE: Will elaborate to specify the problem.

2. *Does the overview list specific facts?*

GG: The overview lists specific facts such as the number of characters and games. An estimate of the number of spells may be useful for non Final Fantasy players: are there 10 different spells or 1000 spells.

RESPONSE: Hard to know how many *Spells* will be included but fair to estimate about 20-60. Will include this in outline.

3. *Does the outline of entity details describe the purpose of each, list attribute data types and constraints and describe relationships between entities?*

GG: The outline of entity details does not include some attribute's data types and does not reflect many attributes presented in the ERD diagram. What is the purpose of the spells attribute presented in the ERD diagram for the Characters entity? The outline mentions M:M relationships but does not explain in detail what the relationships are in the context of the entities.

RESPONSE: True, *Characters* should not have a *Spells* attribute. Will delete. The M:M relationship is elaborated in the *Characters* section but not the *Spells* section. Will repeat text and elaborate where necessary.

LD: Datatypes and constraints are only mentioned in the primary keys in the outline. Some attributes appear in the ER diagram but not mentioned in the outline (e.g. ability_effect in Abilities). Relationships between entities are described.

RESPONSE: Data types will be added for all attributes in text.

4. *Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a logical view of the database?*

GG: The ERD does not represent the three M:M relationships between Action List and Characters, Spells, and Abilities.

RESPONSE: Need to change ERD to reflect the M:M relationships.

There are many attributes presented in the ERD diagram and overview that are not reflected in the other. For example, for the Spells entity, spell_cost is in the ERD but not in the overview and type is in the overview but not in the ERD.

There is no 1:M relationship described in the overview but several are presented in the ERD diagram.

RESPONSE: We need another entity for a 1:M like *Homes*

As another reviewer pointed out, the relationship between Action List and Abilities is 1:M not M:M.

RESPONSE: Disagree, every action can be associated with every ability.

LD: The Action List returns what a character can do. As described in the Abilities entity, a character has only one ability, so the relationship between Action List and Abilities should be 1:M (i.e. each action list can contain 1 ability only), not M:M.

RESPONSE: The confusion is that the action list is a history of actions taken in battle, not a snapshot of options in time. Need to clarify this in description and change entity name to be more precise.

AB: No 1:M relationships are described in the outline, but 1:M relationships are displayed in the ERD for action_list >--|- abilities, action_list >--|- spells, and action_list >--|- characters. I think these relationships were meant to be M:M as described in the outline.

RESPONSE: Yes, need to represent M:M. Currently the ERD is incorrectly showing 3x 1:M out of action_list instead of 3x M:M.

JV: There isn't a 1:M relationship in the outline. There are only 1:1 and M:M relationships in the outline.

RESPONSE: See above, need to add a 1:M relationship.

In the ERD, there seems to be a connection between spells_has_characters and characters but the spell_has_characters is not defined in the outline.
RESPONSE: This is how M:M relationships are shown in Workbench. spell_has_characters is an intersection table not an entity (source "Exploration - The ER Model" video 1 13:45). This is unclear so we will use draw.io in the future.

5. *Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?*

GG: There is an inconsistency in naming between entities and overview. Entities in overview are capitalized while entities in the ERD diagram are lower case. The punctuation of many attributes are different in the overview than in the ERD diagram. For example, spellID is spell_id in the ERD diagram. Entities are plural and attributes are singular.

LD: Capitalization of entity names in the outline but lower case entity names are used in the ERD.

Action List shouldn't have any space in between (the entity name in the outline)

AB:

a. Naming between overview and entity attributes are not consistent. For example, Action List's Actions in the overview vs. action_id in the ERD, and Spells' spellID in the overview vs. spell_id in the ERD.
b. Entities are all plural aside from Action List. Maybe this could be shortened to Actions. Most attributes are singular, however the attributes spells, hit_points, and mana_points are not. I'm not sure the spells attribute is necessary with the M:M relationship and intersection table between Characters and Spells already, and maybe hit_points and mana_points could be expanded to hit_points_value and mana_points_value, respectively.
c. Capitalization is not consistent between the overview and ERD. Entities should be capitalized in the ERD, and the snake_case used in the ERD should be reflected in the overview as well.

JV: There is inconsistency in naming certain entities and attributes since some of them may be written in the outline but others aren't written in the outline and only in the ERD diagram.

Also some names are written differently, for example, in the ERD diagram, you have action_list, where as in the outline it's named Action List with capitalization.

RESPONSE: Feedback correctly states there are several inconsistencies between the outline and ERD due to versioning. Will fix the outline first with our final consensus, and then diagram it with matching formatting. This will ensure consistency. We will choose formatting as follows: entities as "*Characters*" initial capitalization, italicized plural... attributes as "characterName" in camelCase, singular. Instances where we are just talking about the singular entity existing in the table of the same name will be capitalized, but not italicized, for example when we are talking about a Spell in the *Spells* table, it will appear as just demonstrated. Instances where we copy-and-pasted from peer review will remain unaltered, and will be shown as initially typed in ED Discussions.

Actions taken (not inspired by outside feedback):

- change Battle Actions to BattleActions (and all references)- sql doesn't like multiple words
- under Spells, type to spellType- consistency
- varchar to 255 from 250 in all types- memory efficiency
- change Worlds entity to Homes- matches data better
- under Homes, name to homeName- consistency
- add attack max 99 to outline- more clarity for users
- under characters, remove not null from homeID- should be able to add characters without worrying about Home if needed
- remove NOT NULL from BattleActions spell and ability- has to be one or the other, then the other one is NULL
- add actionEffect to Battle Actions- pretty important attribute was missing and can't simply be joined with Spell and Ability effects for reasons elaborated in section below

- add inParty, inBattle to Characters- to author battles and generate reports for only characters participating in battle and to differentiate heroes from villains in the battle
- change post-apocalyptic to postApocalyptic- consistency

Actions NOT taken and why:

There are two interesting relationships in the *BattleActions* entity that may tread the line for a non-normalized relation. However, as elaborated below the entity will be unchanged because they meet criteria.

1. Each Action in the *BattleActions* table has either a Spell OR an Action associated with it, but not both. In some instances this is a transitive dependency because, for example, actionID determines spellID (i.e. Action 1 was to cast “fire”) and that spellID then determines that abilityID is NULL. However, in the event that an ability was used and not a Spell, it is NOT a transitive dependency, for example, if spellID is NULL, spellID does not then determine abilityID because any Ability can be used. Therefore, since it is not ALWAYS a transitive dependency, it would be a mistake to “over-normalize” and create another entity with a composite key as this key would not reliably be a determinant.

2. Each Action in the *BattleActions* table is associated with an “actionEffect”. In an example Action, an actionID determines a spellID, which in its native table has a spellEffect, separate from actionEffect. The reason for this is that, ideally, many more entities are necessary to parse out what exactly the end effect of an Action is.

For example, in this ideal relation, a StatusEffects table has a M:M relationship with characters, and, for example, if a character had a Slow effect active but a Haste was cast on her, the end result would be normal character speed, not 2x speed. Another example is the calculation of effects of *Spells* acting on normal people, or with their own internal logic. A Spell that did damage to a random character is difficult but not impossible to hard code in SQL.

All of these situations have crossed the line into a territory where a design decision to include another programming language would likely be a better choice. Furthermore, for the use case of this database for basic analysis of battles, in which Counts of Spells/Abilities cast and turns taken, that level of detail is unnecessary. For an author of fanfiction this may even be preferable, as the end actionEffect is left to the author of this universe to interpret and apply.

For those reasons we consider *BattleActions* to remain in 3NF.

b) Overview:

In every installment of the wildly popular Squaresoft video games, fans of the studio have come to expect, and adore, a wide host of richly written characters. For the player, these characters are guided through fantastical alien worlds in elaborate soap-opera-like journeys where they evolve, fall in love, and ultimately vanquish their enemies in battle.

In particular, the series *Final Fantasy* contains a particularly extensive universe, spanning 15 core games, 80 spin-off games, two anime series, and a movie. Fans of *Final Fantasy* (and other role playing games) written by Squaresoft easily fall in love with these characters and can end up day dreaming about alternate worlds where these characters interact with each other between their separate games. Sometimes fan-fiction is written, other times entire video games are made with crossovers between the characters. There is no easily accessible database with “stats” from all the characters in one place. This makes it more difficult to program a video game for fans.

Final Fantasy All-Stars is an employee-esque database storage system hosting a variety of characters from the Final Fantasy video game series, which will log the informational data regarding characters amongst the series. Our database will store the statistics of 30 *Characters* across the core 15 games in the series. Each of the *Characters* in the game will be assigned one specific Ability from *Abilities* and a small variety of *Spells* cataloged in their *BattleActions* list. We will be able to summon a specific Character from the database, or a list of *Characters* who know a certain Spell from the *Spells* list.

With this database it is possible to simulate battles with any variety or number of *Characters* with any of their *Spells* and *Abilities* and have those *BattleActions* recorded for posterity. As each *BattleAction* can involve 30 *Characters* * (50 *Spells* + 30 *Abilities*), the *BattleActions* table is a good way to keep the history of actions straight. Analysis on battles can occur to see who contributed more in different battles. Furthermore, anyone who wants to write fanfiction from the battle can look back on the data for inspiration.

Database Outline:

Primary Key / Foreign Key

- **Characters**: records details of 30 Characters. Mana is used to cast spells.
  - characterID: int, auto-increment, unique, not null, PK. Every character will be accounted for with a mandatory ID so there are no instances of a NULL characterID.

    - characterName: varchar(50), not null. Name of the Character.
    - homeID: int, null, FK. Character’s Home from *Homes*. Nullable
    - health: int, not null. How much health does the character have. Max 999.
    - mana: int, not null. How much mana does the character have. Max 99.
    - attack: int, not null. How much basic attack does a character deal to the opponent. Max 99.
    - abilityID: int, null, FK. Links to Character’s unique Ability.
    - inParty: boolean not null, returns true if a Character is actively participating in the battle simulator, and false if the character is not.
    - inBattle: boolean not null, returns true if a Character is the defending hero and false if a Character is the attacking party.
    - **Relationships:** a M:M relationship between *Spells,* as multiple *Spells* can be accessed by several different *Characters* both simultaneously and independently*,* and *Characters* can also access several different *Spells* at the same time. 1:1 relationship between *Abilities* as a specific ability is only available to a specific character. M:0 relationship with *Homes*, as each Home can be the same for several characters. Some Characters don’t have Homes, or otherwise have a Null relationship with Homes. N:M with *BattleActions* through *CharacterActions* intersection table. N:0 with *Spells* through *SpellActions* intersection table.

  - **Homes**: records the background home of each character. Each Home has an ID, name, and post-apocalypse status. Notably, certain *Characters* from a game can be part of the same game/home, but in different stages of apocalypse (usually involves time travel).
    - homeID: int, auto-increment, unique, PK.
    - homeName: varchar(50), not null.
    - postApocalyptic: boolean, not null. Some *Homes* have bitten the dust at some point in time.
    - **Relationships:** 1:M with *Characters*, as a singular Home can be home to many *Characters.*

  - **Abilities**: records the details of *Abilities*, assigning a unique ID to an individual Ability, and ensuring that the Ability will be assigned to a single Character, with no duplicates. A total of 30 *Abilities* will exist in the *Abilities* table. A singular Character will have access to a singular unique Ability from the *Abilities* table. For example, Terra will have Transform, Cloud will have Braver, etc. A specific Ability MUST be assigned to a specific Character, and there is no leftover/unused/NULL ability.

■ abilityID: int, auto-increment, unique, not null, PK.

■ abilityName: varchat(50), not null.

■ abilityEffect: varchar(255) not null. How the ability affects mana, health, status effects, etc.

■ **Relationships:** a 0:1 relationship between *Characters*, as every Character has some unique Ability. However, nullable, in case there are Characters without an Ability. No relationship with *Spells*. N:M with *BattleActions* through *AbilityActions* intersection table.

○ **Spells**: records the available *Spells* that *Characters* may be able to use. Approximately 50 *Spells* will be recorded in *Spells*.

■ spellID: int, auto-increment, unique, not null, PK. Integer to keep track of the different available *Spells*, auto-increment so every different Spell will be accounted for in the list, not NULL as every Spell slot must be used in the database.

■ spellName: varchar(50) not null. Name of the available spell(s). Ex, Thunder, Fire, Revive, Sleep…

■ spellEffect: varchar(255) not null. How the Spell affects mana, health, status effects, etc.

■ spellType: varchar(50) not null. Differentiates the type element/effect the Spell may have. For example, Firaga is a Fire Type Spell, Sleep is a status effect, ect.

■ **Relationships:** As stated earlier, will have a M:M relationship with *Characters.* Each Character could have access to many *Spells*, and every Spell can be used by many different *Characters* through *CharacterSpells* intersection table. N:M with *BattleActions* through *SpellActions* intersection table.

○ **CharacterSpells**: intersection table for Characters and Spells. Characters can use many different types of Spells, and each Spell can be used by many different Characters.

■ characterSpellID: int auto-increment unique not null PK.

■ characterID: int not null FK

■ spellID: int not null FK

■ **Relationships:** N:M with *Characters* through *CharacterSpells* intersection table. Each Character has a set of Spells they are able to cast. Possibly 0 spells (optional).

- **BattleActions**: records and returns the list of what *Characters* can do on its theoretical "turn", should that specific Character need to take a "turn".
  - actionID: int auto-increment unique not null PK.
  - actionEffect: text null, description of an Action's effect in battle.
  - **Relationships:** N:M with *Spells*, *Actions*, and *Characters* through *SpellActions, AbilityActions,* and *CharacterActions* intersection tables, respectively. Each Battle Action is associated with a Character, and a Spell or Action (not both).

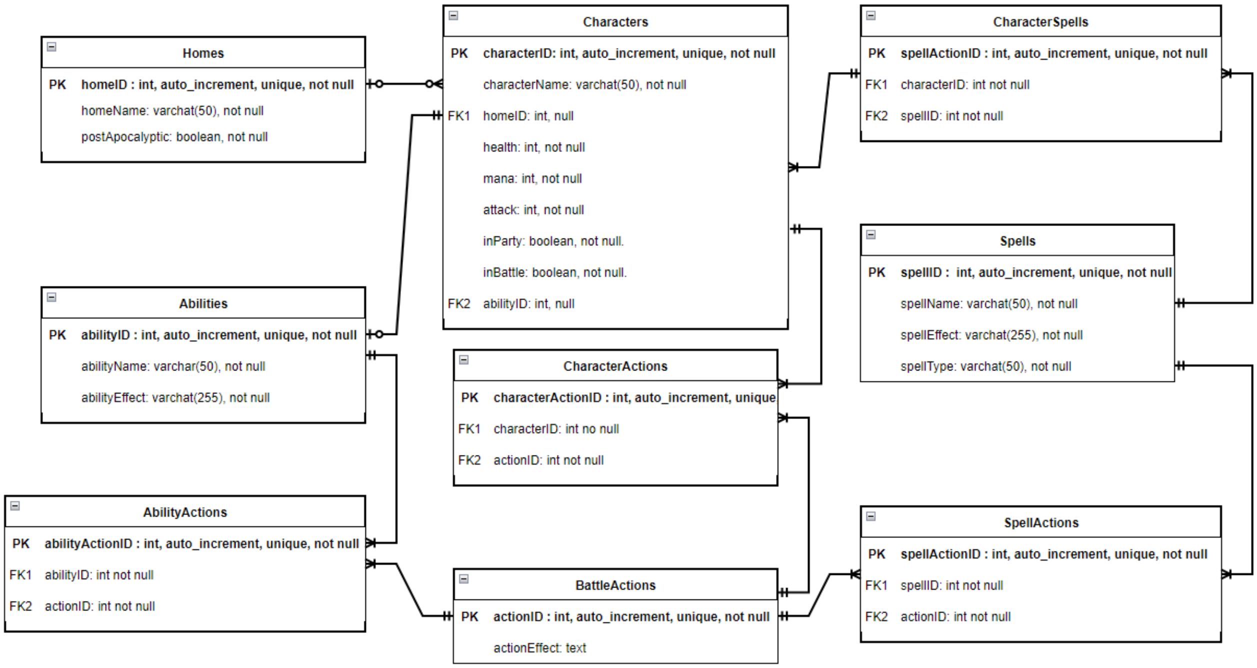
- **SpellActions**: Intersection table for Actions in Battle that are classified as Spells.
  - spellActionID: int auto-increment unique not null PK.
  - actionID: int not null FK. Associated Action.
  - spellID: int not null FK. Associated Spell.
  - **Relationships:** N:M intersection table as each Action may be any Spell and vice versa. Connects *BattleActions* with *Spells*

- **AbilityActions**: Intersection table for Actions in Battle that are classified as Abilities.
  - abilityActionID: int auto-increment unique not null PK.
  - actionID: int not null FK. Associated Action.
  - abilityID: int not null FK. Associated Ability
  - **Relationships:** M:M intersection table as each Action may be any Ability and vice versa. Connects *BattleActions* with *Ability*

- **CharacterActions**: Intersection table for Actions in Battle for different Characters.
  - characterActionID: int auto-increment unique not null PK.
  - actionID: int not null FK. Associated Action.
  - characterID: int not null FK. Associated Character
  - **Relationships:** M:M intersection table as each Action may be

associated with any Character and vice versa. Connects *BattleActions* with *Characters*.

c) Entity-Relationship Diagram:



Homes

PK	**homeID : int, auto_increment, unique, not null**
	homeName: varchat(50), not null
	postApocalyptic: boolean, not null

Characters

PK	**characterID: int, auto_increment, unique, not null**
	characterName: varchat(50), not null
FK1	homeID: int, null
	health: int, not null
	mana: int, not null
	attack: int, not null
	inParty: boolean, not null.
	inBattle: boolean, not null.
FK2	abilityID: int, null

CharacterSpells

PK	**spellActionID : int, auto_increment, unique, not null**
FK1	characterID: int not null
FK2	spellID: int not null

Spells

PK	**spellID : int, auto_increment, unique, not null**
	spellName: varchat(50), not null
	spellEffect: varchat(255), not null
	spellType: varchat(50), not null

Abilities

PK	**abilityID : int, auto_increment, unique, not null**
	abilityName: varchar(50), not null
	abilityEffect: varchat(255), not null

CharacterActions

PK	**characterActionID : int, auto_increment, unique**
FK1	characterID: int no null
FK2	actionID: int not null

AbilityActions

PK	**abilityActionID : int, auto_increment, unique, not null**
FK1	abilityID: int not null
FK2	actionID: int not null

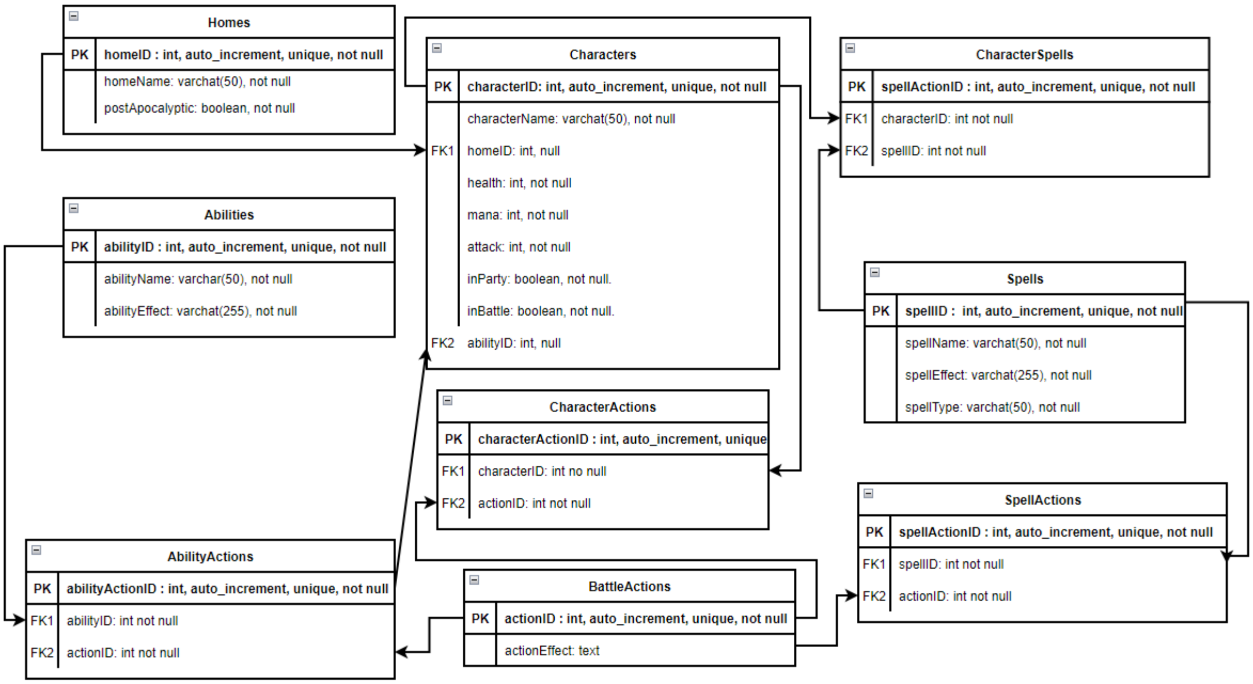
BattleActions

PK	**actionID : int, auto_increment, unique, not null**
	actionEffect: text

SpellActions

PK	**spellActionID : int, auto_increment, unique, not null**
FK1	spellID: int not null
FK2	actionID: int not null

d) Schema:



	Homes
PK	homeID : int, auto_increment, unique, not null
	homeName: varchat(50), not null
	postApocalyptic: boolean, not null

	Characters
PK	characterID: int, auto_increment, unique, not null
	characterName: varchat(50), not null
FK1	homeID: int, null
	health: int, not null
	mana: int, not null
	attack: int, not null
	inParty: boolean, not null.
	inBattle: boolean, not null.
FK2	abilityID: int, null

	CharacterSpells
PK	spellActionID : int, auto_increment, unique, not null
FK1	characterID: int not null
FK2	spellID: int not null

	Abilities
PK	abilityID : int, auto_increment, unique, not null
	abilityName: varchar(50), not null
	abilityEffect: varchat(255), not null

	Spells
PK	spellID : int, auto_increment, unique, not null
	spellName: varchat(50), not null
	spellEffect: varchat(255), not null
	spellType: varchat(50), not null

	CharacterActions
PK	characterActionID : int, auto_increment, unique
FK1	characterID: int no null
FK2	actionID: int not null

	SpellActions
PK	spellActionID : int, auto_increment, unique, not null
FK1	spellID: int not null
FK2	actionID: int not null

	AbilityActions
PK	abilityActionID : int, auto_increment, unique, not null
FK1	abilityID: int not null
FK2	actionID: int not null

	BattleActions
PK	actionID : int, auto_increment, unique, not null
	actionEffect: text

e) Sample Data:

Spells

spellName	spellEffect	spellType
fire	cost 4mp damage single 50hp	fire
firaga	cost 12mp damage all 400hp proc chance to incinerate	fire
ice	cost 4mp damage single 50hp	ice
blizzaga	cost 12mp damage all 400hp proc chance to freeze	ice
lightning	cost 4mp damage single 50hp	lightning
luminaire	cost 12mp damage all 400hp proc chance to blind	lightning
aero	cost 4mp damage single 50hp	wind
aeraga	cost 12mp damage all 400hp proc chance to confuse	wind
quake	cost 4mp damage single 50hp	earth
quaga	cost 12mp damage all 400hp proc chance to stun	earth
cure	cost 4mp restore single ally 100hp	life
esuna	cost 2mp remove negative status effects single char	life
revive	cost 12mp revive single ally 100hp	life
slow	cost 6mp half turn recovery single char	time

Abilities

abilityName	abilityEffect
braver	damage single 3x attack
trance	transform into esper state for 5 turns. 2x attack 2x magic def
black hole	send an enemy to the nether realm
electrocute	stun and damage all electronic enemies 400
apocalypse	damage all heroes 700

Homes

homeName	postApocalyptic
Nibelheim	false
Esper World	false
Zeal	false
Proto Dome	true
Unknown	false

Characters

characterID	characterName	homeID	attack	health	mana	inParty	inBattle	abilityID
1	Cloud Strife	Nibelheim (subquery for homeID)	90	999	99	true	true	subquery for abilityID
2	Terra	Esper World (subquery for homeID)	64	999	99	true	true	subquery for abilityID
3	Magus	Zeal (subquery for homeID)	82	999	99	true	true	subquery for abilityID
4	Robo	Proto Dome (subquery	96	999	99	false	false	subquery for abilityID

		for homeID)						
5	Lavos	Unknown (subquery for homeID)	99	999	99	false	true	subquery for abilityID

CharacterSpells

characterID	spellID
1	subquery for spellID matching ‘lightning’
1	subquery for spellID matching ‘cure’
2	subquery for spellID matching ‘firaga’
2	subquery for spellID matching ‘revive’
3	subquery for spellID matching ‘luminaire’
3	subquery for spellID matching ‘slow’
5	subquery for spellID matching ‘quaga’

BattleActions

actionID	actionEffect
1	all heroes damaged 700hp
2	heal Terra 100hp
3	Terra 2x attack 2x magic def for 5 turns

SpellActions

actionID	spellID
2	subquery for spellID where spellName = ‘cure’

AbilityActions

actionID	abilityID
1	subquery for abilityID where abilityName = ‘apocalypse’

3	subquery for abilityID where abilityName = ‘trance’

CharacterActions

actionID	characterID
1	subquery for characterID where characterName = ‘Lavos’
2	subquery for characterID where characterName = ‘Cloud Strife’
3	subquery for characterID where characterName = ‘Terra’