1.

To prove that $Eagg$ = 1/M $Eavg$, under the given assumptions, let's start by expanding the expression for $Eagg(x)$:

$Eagg(x)$ = E[({1/M∑$\epsilon i(x)$}^2)]
Now, we can expand the square:

$Eagg(x)$ = E[({1/M^2∑$\epsilon i(x)$∑$\epsilon j(x)$})]
Applying the expectation operator:

$Eagg(x)$ = 1/M^2 ∑∑$E(\epsilon i(x)\epsilon j(x))$

Now, let's apply the given assumptions:

Each of the errors has a 0 mean:

$E(\epsilon i(x))$ = 0 for all i
Errors are uncorrelated:

$E(\epsilon i(x)\epsilon j(x))$ = 0 for all i ≠ j
Using these assumptions, we can simplify the expression:

$Eagg(x)$ = 1/M^2 ∑$E(\epsilon i(x)$^2)

Now, we can rewrite the expression for $Eavg$ using the given definition:

$Eavg$ = 1/M ∑$E(\epsilon i(x)$^2)
Comparing the expressions for $Eagg(x)$ and $Eavg$, we can see that:

$Eagg(x)$ = 1/M $Eavg$

Thus, under the given assumptions of 0 mean errors and uncorrelated errors, the expected squared error of the aggregated model is equal to 1/M times the average expected squared error of the individual models.

2.

E_agg: Expected error of the aggregated model (a model that is a linear combination of M models)
E_av: The average expected error of individual models
$\epsilon i(x)$: Error of the i-th model on a given input x
λi: The weight associated with the i-th model (λi ≥ 0 and $\sum\lambda i$ = 1)

$Eagg$ = $E(\sum\lambda i\epsilon i(x))$ and Eav = M^(-1)$\sum E(f(xi))$. We want to show that $E(\sum\lambda i\epsilon i(x))$ ≤ M^(-1)$\sum E(f(xi))$.
Let's use Jensen's inequality, which states that for any convex function f, f($\sum\lambda ixi$) ≤ $\sum\lambda if(xi)$, where xi are real numbers and λi are non-negative real numbers such that $\sum\lambda i$ = 1.

f($\sum\lambda i\epsilon i(x)$) ≤ $\sum\lambda if(\epsilon i(x))$

($\sum\lambda i\epsilon i(x)$)^2 ≤ $\sum\lambda i(\epsilon i(x))$^2

E(($\sum\lambda i\epsilon i(x)$)^2) ≤ E($\sum\lambda i(\epsilon i(x))$^2)

Since the left side represents the expected error of the aggregated model squared, it can be denoted as
$Eagg$^2.

On the right side, we can exchange the order of summation and expectation, as expectation is a linear operation:

$Eagg$^2 ≤ $\sum\lambda iE((\epsilon i(x))$^2)

Now, note that $E((\epsilon i(x))$^2) is the expected squared error of the i-th model, which is related to the expected error of the i-th model.

Since M^(-1)$\sum E(f(xi))$ represents the average expected error, and λi are the weights, we can rewrite the inequality as:
$Eagg$^2 ≤ $\sum\lambda iE((\epsilon i(x))$^2)

We can rewrite M^(-1)$\sum E(f(xi))$ as the weighted sum of expected squared errors of the individual models:
Eav = M^(-1)$\sum E((\epsilon i(x))$^2)

Now, multiply both sides of the inequality by M:

$MEagg^2 \leq \sum \lambda i E((\epsilon i(x))^2)$

Since $MEav = \sum \lambda i E((\epsilon i(x))^2)$ and $M > 0$, we can compare the left sides of the two inequalities:

$MEagg^2 \leq MEav$

Divide both sides by M:
$Eagg^2 \leq Eav$

Since both $Eagg$ and Eav are non-negative (errors are non-negative), we can take the square root of both sides:
$\sqrt{Eagg^2} \leq \sqrt{Eav}$

$Eagg \leq Eav$

3.

First, let's define the overall training error at the end of T steps, E:
$E = (1/N) * \Sigma_i [H(x_i) \neq y_i]$
where N is the total number of data points, x_i is the ith data point, y_i is the true label for x_i, and $H(x_i)$ is the final hypothesis for x_i.
Now, let's rewrite $H(x_i)$ using the equation provided:
$H(x_i) = \text{sign}(\Sigma^{t=_1 \text{to} T} \alpha^t * \hbar^t(x_i))$
Since the weight for the point i at step t+1 is given by:
$D_{\square+1}(i) = D_\square(i) / Z_\square * \exp(-\alpha^t * \hbar_\square(i) * y(i))$
We can compute the upper bound for the training error at each step as follows:
$\Sigma_i [D_\square(i) * [\hbar_\square(x_i) \neq y(x_i)]] <= \exp(-2 * \gamma_\square^2)$
Next, let's express the overall training error E in terms of the weights D:
$E = \Sigma_i [D_\square(i) * [H(x_i) \neq y_i]] / N$
Now, let's use the fact that the weights D are normalized at each step (i.e., $\Sigma_i D_\square(i) = 1$):
$E = \Sigma_i [D_\square(i) * [H(x_i) \neq y_i]] / \Sigma_i D_\square(i)$
We can now upper bound the overall training error E by the sum of the upper bounds of the training error at each step:
$E <= \Sigma_i [D_\square(i) * \exp(-2 * \gamma_\square^2)] / \Sigma_i D_\square(i)$
Since $\Sigma_i D_\square(i)$ is a normalization factor that ensures the weights sum up to 1, we can simplify the expression:
$E <= \exp(-2 * \Sigma_{\square=_1}^{\text{to} T} \gamma_\square^2)$

Thus, we have proven that at the end of T steps, the overall training error E will be bounded by:
$\exp(-2 * \Sigma_{t=1}^{\text{to}T} \gamma_t^2)$